NR INDEKSU 353965
PL ISSN 0860 – 1674

10

Z MIKROKOMPUTEREM NA TY

# Bajtek

MIESIĘCZNY DODATEK DO „SZTANDARU MŁODYCH"
NR 10(22) PAŹDZIERNIK 1987 CENA 100ZŁ




BREAK DANCE KUBUSIA

FORTH

BASIC XL

WIRTUOZ

ALIENS

# KRUSZENIE BARIER

*Tematem dnia jest niewątpliwie rozpoczęcie drugiego etapu reformy gospodarczej. I to nie dlatego wcale, że zaczęło się zapowiadane od dawna likwidowanie i łączenie szeregu ministerstw i urzędów centralnych. Owszem, to też jest ciekawe, ale głównie dla samych zainteresowanych. Natomiast dla znakomitej większości naszego społeczeństwa wejście w drugi etap reformy to szansa może nie na skokowe jeszcze, ale w każdym razie szybsze niż do tej pory wychodzenie z kryzysu. A to już jest dla każdego gra warta świeczki!*

*Nadchodzą czasy kiedy najbardziej liczyć się będzie aktywność, przedsiębiorczość, pomysłowość. Czyli liczyć się będzie to wszystko, o co od dawna i w „Bajtku", i w „Sztandarze Młodych" zabiegaliśmy i walczyliśmy. Tak się zresztą składa, że rynek komputerowy jest już rynkiem z prawdziwego zdarzenia (to samo można zresztą też powiedzieć o rynku pism komputerowych). Jesteśmy więc jak gdyby do nowych reguł gry ekonomicznej przygotowani lepiej niż inni.*

*Ale stworzenie nowych mechanizmów regulujących naszą gospodarkę to jeszcze nie wszystko. Potrzebni są bowiem ludzie, którzy będą chcieli i umieli z tych mechanizmów skorzystać. I to ludzie zatrudnieni we wszystkich sektorach naszej gospodarki. Dlatego zgadzam się z poglądem, że właśnie w tym obszarze – powiedzmy: kadrowym – rozstrzygnie się sukces lub przegrana naszej reformy gospodarczej.*

*„Wszystko zależy od człowieka – stwierdził na ostanim V Plenum Komitetu Centralnego Polskiej Zjednoczonej Partu Robotniczej towarzysz Wojciech Jaruzelski. – Kadry, ich przekonanie i zdecydowanie w realizacji przyjętej linii będą w ostatecznym rachunku najważniejsze... w szerokim „manewrze kadrowym" wynikającym z procesu reorganizacji najlepsi ludzie powinni znaleźć się na stanowiskach, gdzie decydować się będzie powodzenie reformy. Ludzie z twórczą wyobraźnią, z konsekwencją, z odwagą. Boiwiem słaby pracownik może spowodować, że nawet dobre mechanizmy nie będą skuteczne. Natomiast dobry, nawet przy niedoskonałych mechanizmach potrafi być efektywny, twórczy."*

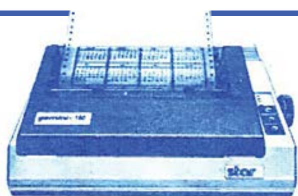
*I zaraz potem Wojciech Jaruzelski dodał: „Chodzi w szczególności o to, aby zapewnić operatywne systemowe eliminowanie nieudolnych i śmiałe wysuwanie skutecznych, zdolnych... w kreowaniu decyzji kadrowych musi być coraz większy, autentyczny udział czynnika społecznego, kolektywów. Właśnie oddolna inspiracja oraz społeczna aprobata dla słusznych decyzji kadrowych jest sprawą niezwykłej wagi." Przypominam te słowa, gdyż w swoisty sposób adresowane są one przede wszystkim do ludzi młodych. Podczas różnych spotkań i dyskusji często słyszymy na spotkaniach o barierach, na jakie napotyka młode pokolenie przy każdej próbie zrobienia czegoś nowego, oryginalnego, twórczego. Nikt oczywiście nie obiecuje (bo nie może tego obiecać), że tych barier teraz nie będzie. Będą, może nawet większe niż poprzednio. Ale warto wiedzieć, że tak wyraźnie wyłożona najwyższa wola polityczna w naszym kraju jest po stronie aktywności, twórczości, innowacyjności. Rzeczą młodego pokolenia jest wykazanie teraz takich kwalifikacji i umiejętności nie zrażania się trudnościami, aby wygrać swą „grę o jutro". Pewne jest, że nigdy jeszcze reguły gry tak bardzo nam nie sprzyjały!*

*Aktywność i prężność całego społeczeństwa będzie teraz w większym niż uprzednio stopniu sumą aktywności i prężności indywidualnych. Jedni próbują nazywać to nowym NEP-em, inni „pierestrojką po polsku", inni jeszcze inaczej. Myślę, że tym razem na szczęście nie w słowach problem – tylko w treści. A ta akurat będzie taka, jaką sobie sami stworzymy i wywalczymy. Bo uchwały sejmowe i inne akty prawne mogą tylko nakreślić ogólne ramy tego co wolno i nie wolno („dozwolone jest wszystko czego nie zabrania prawo"). Natomiast od nas zależy czy wypełnimy te ramy konkretną treścią, czy w pełni skorzystamy z szans, jakie te nowe ramy stwarzają? Zgodnie z właściwym mi optymizmem uważam, że tak będzie, że uda się zwiększyć i skoordynować wysiłek większej niż dotychczas liczby obywateli PRL. A zwiększenie trafnie ukierunkowanego wysiłku indywidualnego to zarazem większy wysiłek, a więc również i większy efekt w skali społecznej.*

*Zniesienie barier krępujących inicjatywę – bo to chyba należałoby uznać za istotę zmian w naszym systemie gospodarczym – stawia nas wszystkich przed nowym wyzwaniem. Czy będziemy umieli skorzystać z możliwości, o które tak długo zabiegaliśmy? Mogę w imieniu redakcji „Bajtka" odpowiedzieć, że postaramy się tak właśnie dalej działać!*

*Waldemar Siwiński*

*PS Obiecałem miesiąc temu poinformować Czytelników o naszych działaniach w zakresie stworzenia pierwszego międzynarodowego pisma komputerowego adresowanego do młodzieży. Z rozmów przeprowadzonych na początku października w Komitecie Centralnym Komsomołu wynika, że również naszym radzieckim kolegom bliskie są takie idee. Wkrótce poinformujemy więc o sprawie bardziej szczegółowo!*



## BAJTEK WE WŁOSZECH!



**Bajtkowa ekipa z Warszawskiego Pałacu Młodzieży pozdrawia ze słonecznej Italii Wszystkich Czytelników!**

# WYBIERZ SAM




**BAJTEK" — MIESIĘCZNY DODATEK DO „SZTANDARU MŁODYCH"**
**ADRES**: 00-687 Warszawa, ul. Wspólna 61. Tel 21-12-05. Przewodniczący Rady Redakcyjnej: Jerzy Domański – -redaktor naczelny „Sztandaru Młodych".
**ZESPÓŁ REDAKCYJNY**: Waldemar Siwiński (z-ca re daktora naczelnego „SM" — kierownik zespołu „Bajtka"), Roman Poznański (z-ca sekretarza redakcji „SM" — sekretarz zespołu „Bajtka"), Krzysztof Czernek, Sławomir Gajda (red. techniczny), Andrzej Gogolewski, Andrzej Kowalewski, Andrzej Podulka, Sławomir Polak, Wanda Rosz kowska (opr. graficzne), Kazimierz Treger, Marcin Waligórski, Roman Wojciechowski. Zdjęcia w numerze Leopold Dzikowski.
**Klany redagują:**
Commodore — Klaudiusz Dybowski, Michał Silski,
Amstrad-Schneider — Tomasz Pyć Sergiusz Wolicki
Spectrum — Konrad Fedyna, Michał Szuniewicz,
Atari — Wiesław Migut, Wojciech Zientara.
Fotoskład — Tadeusz Olczak
Montaż offsetowy — Grażyna Ostaszewska,
Korekta — Maria Krajewska, Ewa Mowińska.
**WYDAWCA**: RSW „Prasa-Książka-Ruch" Młodzieżowa Agencja Wydawnicza, al. Stanów Zjednoczonych 53 04-028 Warszawa Telefony: Centrala 13-20-40 do 49, Redakcja Reklamy 13-20-40 do 49 w. 403, 414.
Cena 100 zł.
Skład techniką CRT-200, przygotowalnią offsetowa i druk: PRASOWE ZAKŁADY GRAFICZNE RSW „PRASA-KSIĄŻKA-RUCH" w Ciechanowie, ul. Sienkiewicza 51. Zam. nr 072247 nakład 250000 egz., K-109




Bajtek

# WIRTUOZ

## Rozmowa z Markiem Bilińskim – kompozytorem

***— Na estradzie występuje muzyk otoczony masą różnych instrumentów elektronicznych. Najpierw stroi to wszystko godzinę, a im więcej sprzętu i im bardziej skomplikowane przygotowania — tym większy szacunek publiczności. Później występuje w słuchawkach na uszach i... nie bardzo wiadomo, co gra on, a co grają jego instrumenty.***

— Wszystko jest dziełem muzyka i zależy od jego woli. Przecież to on zaprogramował te urządzenia, dobrał barwę dźwięków, tempo, czas utworu. On utwór skomponował, a elektroniczna technika tylko pomogła go wykonać.

***— Skomponował przesuwając potencjometry i naciskając przyciski?***

— Tak się może wydawać na pierwszy rzut oka. W teledysku „Ucieczka z tropiku" bawię się joystickiem i suwakami. Wystarczy jeden ruch ręką i... przeciwnik wylatuje w powietrze. A wszystko załatwia wszechmocny komputer. Na taśmie video zmuszam komputer do wszystkiego. Szkoda tylko, że w rzeczywistości w żaden sposób nie udaje się go zmusić do komponowania...

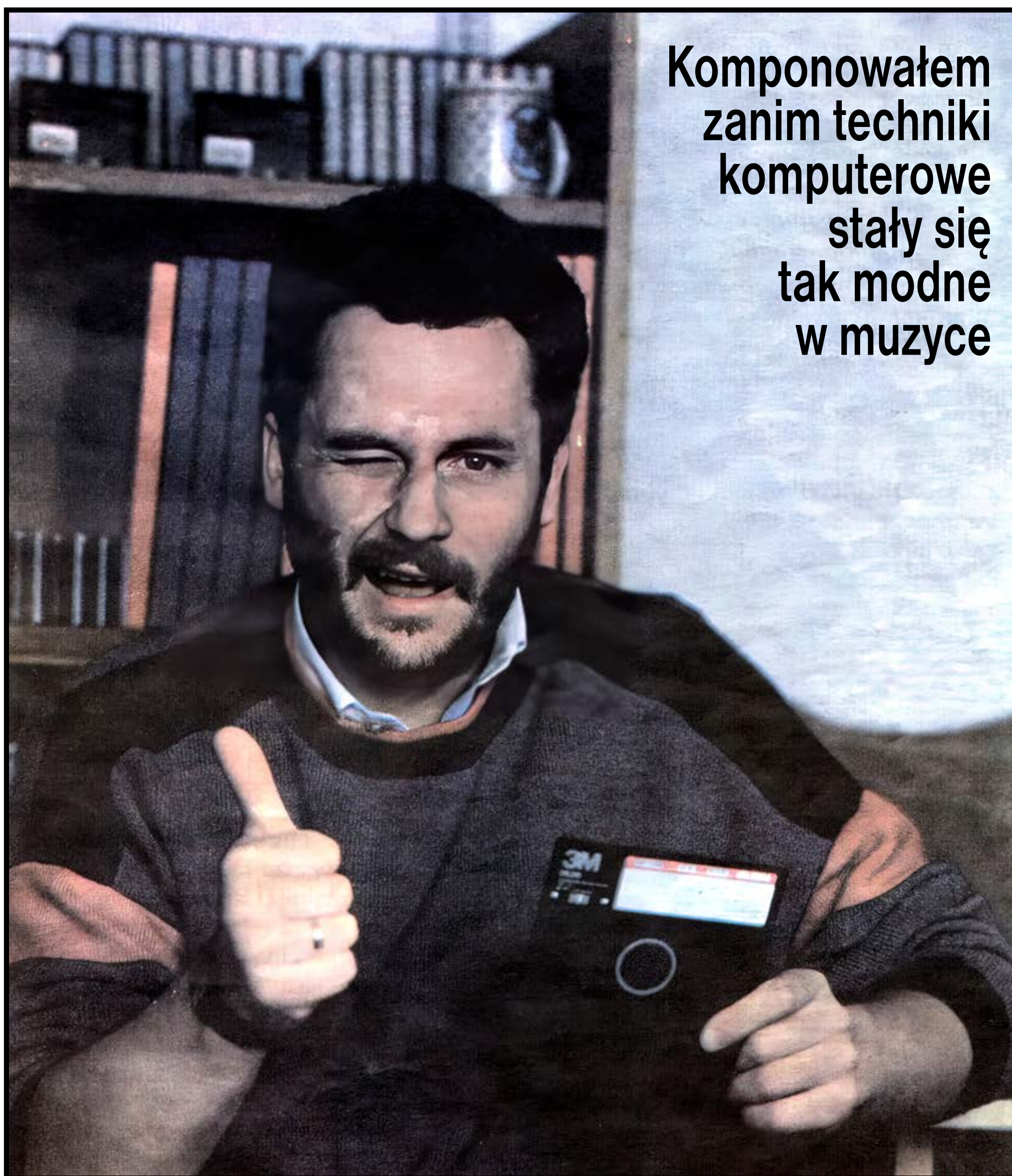
***— W takim razie do czego jest potrzebny? Do szpanowania na estradzie?***

— Przede wszystkim wyjaśnijmy podstawową kwestię: tworzę muzykę, którą potocznie nazywa się elektroniczną, ale wcale nie używam do tego komputera personalnego, czy nawet typowego profesjonalnego. W swoim elektronicznym instrumentarium najczęściej posługuję się sekwencerem QX1 firmy Yamaha. Jest to komputer przystosowany do zabiegów kompozycyjnych. Każdą sekwencję czy też każdy pomysł kompozytorski zamiast zapisywać na taśmie magnetofonowej, wprowadzam do pamięci sekwencera. Wszystko to zapisuje on w formie informacji, w które w każdej chwili mogę ingerować — w odróżnieniu od posługiwania się taśmą magnetofonową. Mówiąc najprościej: mogę jakiś fragment utworu wydłużyć lub skrócić ewentualnie usunąć. W pracy kompozytorskiej jest to duża pomoc, mogę poszczególne frazy rozbudowywać, upraszczać lub w zależności od potrzeby w odpowiednim fragmencie utworu powielać. Wszystko bez nagrywania na taśmę. Ma to duże znaczenie w komponowaniu muzyki filmowej, gdzie bardzo ważna jest synchronizacja obrazu z dźwiękiem. Sekwencer może minimalnie zwolnić lub przyspieszyć tempo podkładu muzycznego dostosowanego do długości sceny.

***— Wystarczy więc wydać odpowiednie polecenia i... maszynka sama pracuje. Wygodnictwo!***

— I nie widzę w tym nic zdrożnego. Po to kupiłem sprzęt, żeby mi ułatwiał pracę. Przecież musiałbym wziąć gumkę i ołówek oraz przepisywać rękopis, robić repetycje. Maszyna nie wyręcza mnie w myśleniu, tylko pomaga w mechanicznych czynnościach. Komponując utwór i tak piszę partyturę — tego nie da się uniknąć. Nawet, gdybym używał specjalnego programu — powiedzmy edytora nut — na ekranie monitora powstałby klasyczny zapis nutowy.

Komponowałem, zanim techniki komputerowe stały się tak modne w muzyce. Mam wyższe wykształcenie muzyczne — kończyłem Wydział Instrumentalny Akademii Muzycznej w Poznaniu. Gram na fortepianie, kontrabasie i klarnecie, znam zasady gry na perkusji. Umiejętność gry na tych instrumentach pozwala mi wczuć się w rolę muzyków.




Sekwencer może posługiwać się ośmioma instrumentami i oprócz wspomnianych zabiegów kompozytorskich — może pełnić podczas koncertu rolę dyrygenta moich elektronicznych muzyków. Gdy on prowadzi wszystkie moje instrumenty, ja mogę zająć się tylko improwizacją.

***— Można więc stosować programy typu edytor tekstów?***

— Tak, ale dotyczą one pięciolinii. Sekwencer jest szalenie zdolnym i zdyscyplinowanym muzykiem — mówiąc żartobliwie — albo bardzo pracowitym kopistą.

***— Skoro jest taki zdolny, to może w przyszłości nauczy się komponować? Zwolennicy sztucznej inteligencji wierzą, że maszyny osiągną taki poziom sprawności, iż będą mogły samodzielnie wykonywać nawet złożone operacje logiczne.***

— Złożone operacje logiczne tak, ale nie złożone operacje myślenia abstrakcyjnego a przecież każda sztuka jest abstrakcją.

***— Mówi pan to jako kompozytor, aby nie podważać swojej roli. A może w przyszłości będzie możliwe wydanie polecenia, aby komputer skomponował utwór muzyczny.***

— Dobrze, puśćmy wodze fantazji. Wyobraźmy sobie takie zadanie: skomponować utwór w stylu Debussy'ego lub Ravela, o długości pięciu minut. I co dalej? Wpisanie w taki program wymogów stylu, czyli harmonii, instrumentacji, rytmiki i tak dalej spowodowałoby, że praktycznie trzeba byłoby taki utwór skomponować i... przenieść na język programu, na którym dany komputer pracuje.

Poza tym, tak jak na dłoni linie papilarne są niepowtarzalne, tak osobowość każdego twórcy jest inna. Tych subtelnych różnic nawet najinteligentniejszy komputer nie jest w stanie wychwycić. Uważam, że technika komputerowa w muzyce może doskonale pełnić funkcje pomocnicze np. tworzyć nowe barwy, rozwijać inwencję, ze względu na swoją pamięć cyfrową, ułatwi stworzenie świetnej biblioteki pomysłów. To wszystko, a reszta zależy od talentu artysty. Warto, aby o tym pamiętali młodzi pasjonaci komputerów i informatyki.

*Rozmawiał:*
*Roman Wojciechowski*


http://www.t2e.pl/reduksy

# PROGRAMOWAĆ MOŻE KAŻDY



## PODSTAWY PROGRAMOWANIA W JĘZYKU

# FORTH

**FORTH jest językiem wysokiego poziomu, jednak udostępnia środki bliskie językowi maszynowemu i dlatego traktowany jest jako język assemblerowy. Program w języku FORTH charakteryzuje sie dużą szybkością działania i małym wykorzystaniem pamięci operacyjnej.**

Podczas naszego kursu zapoznamy Czytelników z podstawami programowania w języku Fig-FORTH, którego implementacje są dostępne na większości komputerów domowych. Przykłady będą wykonane na mikrokomputerze ZX SPECTRUM, oraz na programie umożliwiającym użytkowanie FORTH opracowanym w **firmie Hbersoft.** Budowa ich będzie taka, aby bez kłopotu mogły być odtworzono na innym typie komputera, a konstrukcje dotyczące tylko SPECTRUM będą wyraźnie zaznaczone.

W trybie bezpośrednim FORTH (podobnie jak BASIC) działa jak kalkulator, z tą tylko różnicą, że dane są wprowadzane w odwrotnej notacji polskiej. Dane wejściowe są wyświetlane na ekranie i jednocześnie zapamiętywane w osiemdziesięcioznakowym buforze wejściowym. Gdy jest on całkowicie zapełniony, lub gdy wciśniesz ENTER (RETURN), aby zaznaczyć zakończenie wprowadzania z klawiatury, komputer rozpoczyna przeszukiwanie bufora, zatrzymując się na pierwszym napotkanym znaku spacji. Następnie łańcuch alfanumeryczny, który został zidentyfikowany przed znakiem spacji, jest porównywany z listą słów, jakie znajdują się w słowniku. Można ją zobaczyć po wprowadzeniu VLIST i wciśnięciu **Enter**. Jeżeli maszyna rozpozna wprowadzone słowo jako jedno z tych, które są w słowniku, zostaje wykonane związane z nim działanie (w przypadku VLIST jest to wyświetlenie zawartości słownika). Jeżeli wprowadzona wielkość jest liczbą, zostaje ona odłożona na stos. Jeżeli nie jest to ani liczba, ani słowo ze słownika, zostanie wyświetlony komunikat o błędzie.

Strukturę danych nazywaną stosem można porównać ze stertą książek poukładanych na stole, jedna na drugiej. W danej chwili możemy wziąć tylko tę książkę, którą odłożyliśmy ostatnio, jeżeli chcemy dostać książkę położoną niżej, musimy najpierw zdjąć odpowiednią liczbę książek z góry.

Wprowadźmy następujący ciąg znaków, pamiętając o pozostawieniu spacji nimi (spacja pełni rolę separatora):

```
4 3 * .
```

W odpowiedzi otrzymamy liczbę 12. Sposób, w jaki pracował komputer, jest następujący: rozpoznał pierwszą wielkość- występującą przed znakiem spacji (w tym wypadku 4). Ponieważ nie jest to słowo, lecz liczba, zostaje ona odłożona na stosie. Podobna procedura została powtórzona dla 3. Następnie FORTH odczytał gwiazdkę i zinterpretował ją jako słowo ze słownika i, ponieważ pracuje w trybie bezpośrednim, wykonał.

Gwiazdka oznacza: weź ze stosu liczbę, która znajduje się na jego szczycie (3) oraz drugą również ze stosu (4), pomnóż je przez siebie,


http://www.t2e.pl/reduksy

a wynik połóż z powrotem na stos. Na stępnym słowem, jakie zinterpretował FORTH jest kropka. Powoduje ona pobranie liczby ze szczytu stosu i wyświetlenie jej na ekranie. Spróbujmy rozszerzyć nasz przykład:

**4 3 * 13 + .**

Az do * wszystko przebiega tak samo, jak wyżej i liczba 12 znajduje się na szczycie stosu. Następnie na stos zostaje odłożona liczba 13. Znak plus jest także słowem, FORTH — powoduje dodanie do siebie dwóch liczb ze stosu i odłożenie tam wyniku. Kropka, jak poprzednio, wyświetla liczbę ze stosu. W naszym przypadku jest to 25. Nie jest konieczne wprowadzanie wszystkich danych w jednej linii. Można to robić wciskając za każdym razem ENTER. W przypadku, gdy wprowadzimy zbyt dużo znaków (ponad 80), pojawi się komunikat o błędzie, ponieważ FORTH nie rozpozna prawidłowo ostatniego słowa. Ponieważ zostały wykonane wszystkie działania związane ze słowami występującymi wcześniej, możemy kontynuować wprowadzanie od wskazywanego przez komunikat maszyny.

Niektóre słowa znajdujące się w słowniku nie mogą być używane w trybie bezpośrednim, a próba ich wprowadzenia spowoduje błąd. Słowa te mają zastosowanie w trybie kompilacji, w którym możemy definiować nowe słowa, a więc rozszerzać słownik. A oto przykład:

**: PRZYKL1 * + . ;**

Dwukropek instruuje system o tym, że wszystkie słowa i liczby występujące dalej, aż do średnika, mają być skompilowane, a rezultat określony przez słowo PRZYKL1. Wprowadźmy:

**13 4 3 PRZYKL1**

rezultatem jest jak w poprzednim przykładzie — 25. PRZYKL1 określa mnożenie, dodawanie i wyświetlenie rezultatu, funkcje jakie wcześniej wprowadziliśmy jako oddzielne słowa.

Zdefiniujmy kolejne słowo:

**:PRZYKL2 13 4 3 PRZYKL1;**

Wynikiem działania będzie wyprowadzenie liczby 25. PRZYKL2 pokazuje jak można, wykorzystując hierarchiczną budowę FORTH-a, definiować nowe słowa, poprzez wykorzystanie już istniejących, tak że w końcu cały program będzie mógł być wywołany przez wprowadzenie jednego słowa. Program w języku FORTH nadaje każdemu blokowi funkcjonalnemu nazwę, ich grupy są identyfikowane przez kolejne nazwy i.t.d. Taka struktura wymaga zaplanowania z góry tego, jakie program będzie miał części i co w każdej z nich będzie wykonywał.

Programista piszący w języku FORTH powinien poznać różne systemy liczbowe (w szczególności binarny). Przedstawimy na przykładzie zasady zapisu liczb w postaci dwójkowej:

Oznaczamy wagi poszczególnych cyfr liczby. Pierwsza od prawej 2↑0 (2 do potęgi 0), druga 2↑1 itd. Następnie mnożymy wszystkie cyfry przez ich wagi, dodajemy rezultaty i otrzymujemy wartość liczby w systemie dziesiętnym. Na przykład:

$110001_2 = 2^5\times1 + 2^4\times1 + 2^3\times0 + 2^2\times0 + 2^1\times0 + 2^0\times1 = 32 + 16 + 0 + 0 + 0 + 1 = 4910$

Podobny algorytm można powtórzyć dla dowolnego systemu liczbowego, z tym, że wagi będą mieć postać:

$(\text{podstawa systemu})^{\text{wykładnik}}$

gdzie wykładnik ma postać analogiczną jak wyżej, na przykład dla systemu trójkowego mamy:

$3^0, 3^1, 3^2$ itd.

Liczba $2101_3 = 54 + 9 + 1 = 64_{10}$

W systemach o podstawie większej od dziesięciu cyfry 10, 11 i większe zastępujemy literami. Na przykład w systemie heksadecymalnym (szesnastkowym) mamy następujące cyfry:

0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F,

## REPREZENTACJA LICZB

Wszystkie dane, na jakich operuje FORTH, traktowane są jako słowa 16-to bitowe.

Znaczenie poszczególnych słów nie jest rozróżnialne przez komputer i dlatego koniecznym staje się poznanie różnych sposobów zapisu liczb, jakie mogą być używane podczas posługiwania się FORTHEM. Liczby ze znakiem i bez znaku:

Szesnastobitowe słowo może przyjmować wartości od 0 do 65535. Bit n-ty ma wartość dwa do potęgi n, a najmniej znaczący bit oznaczamy jako zerowy. Istnieje również drugi sposób zapisu liczb. Na wartość bezwzględną wpływają bity od 0 do 14, a bit 15 określa znak liczby (jest to bit znaku). Jeżeli bit 15 jest zerem, słowo przyjmuje wartości od 0 do 32767. Jeżeli jest jedynką, liczba pochodzi z przedziału od -32768 do -1. Widzimy, że liczba dodatnia (lub równa zero), gdy bit 15 jest zerem, a ujemna, gdy jest jedynką. Wprowadźmy:

**40000 .**

wyświetloną liczbą nie będzie 40000, a - 25536. Dzieje się tak dlatego, ponieważ obydwie te liczby są reprezentowane przez 1001110001000000. Binarny zapis liczby 40000 został odczytany jako liczba ze znakiem i wyświetlone zostało -25536. FORTH pozwala używać zarówno zapisu czysto binarnego, jak i zapisu ze znakiem. Kropka powoduje, że liczba jest odczytana w postaci „ze znakiem", natomiast U. interpretuje ją jako binarną bez znaku.

**Liczby podwójnej precyzji:**

Słowa szesnastobitowe mają często niewystarczający zakres i dlatego konieczne staje się operowanie na słowach 32-bitowych. Są one przedstawiane jako dwa słowa 16-bitowe, położone na stosie jedno na drugim w taki sposób, że bardziej znacząca część znajduje się ponad mniej znaczącą. 32-bitowe słowo może przyjąć 4 294 867 206 różnych wartości, a zapis ze znakiem pozwala osiągnąć zakres + /-2147433603. Jeżeli chcemy położyć na stosie liczbę podwójnej precyzji, należy umieścić, gdziekolwiek w niej kropkę. Na przykład 5.00 i 500. będą potraktowane jako liczba 500 i położone na stosie w podwójnej precyzji. Wyświetlenie liczby podwójnej precyzji następuje poprzez D. (w reprezentacji „ze znakiem").

Uwaga: Wprowadzenie liczby spoza zakresu +/- 32767 bez znaku kropki spowoduje zinterpretowanie jej jako podwójnej precyzji, lecz jej bardziej znacząca część zostanie odrzucona, a na stosie zostanie położona tylko mniej znacząca część.

**Praca w innych systemach liczbowych :**

Standardowo FORTH przyjmuje i wyprowadza liczby w postaci dziesiętnej, ale możliwa jest praca w innych systemach. Słowo **HEX** wprowadza system szesnastkowy, natomast **DECIMAL** powoduje powrót do systemu dziesiętnego. Można pracować także w innych systemach. Słowo ! powoduje umieszczenie drugiego elementu ze stosu na miejscu określonym przez pierwszy. Sekwencja:

**n BASE !**

nadaje zmiennej BASE wartość n.

Zmienna ta określa bazę (podstawę) systemu w jakim będą wprowadzane i wyprowadzane liczby.

Uwaga: BASE określa adres pamięci, pod jakim jest schowana wartość zmiennej.

**HEX** odpowiada **16 BASE !**, natomiast **DECIMAL** — **10 BASE !** Praca w systemie dwójkowym jest osiągalna przez **2 BASE !**.

UWAGA: Jeśli chcemy powrócić z systemu dwójkowego do dziesiętnego wprowadzenie **10 BASE !** nie spowoduje żadnej zmiany, ponieważ 10 to binarne 2., a właśnie w takim systemie pracujemy. Należy wprowadzić **1010 BASE !**, ponieważ właśnie

$1010_2 = 10_{10}$

Słowo @ powoduje odczytanie zawartości pamięci o adresie wskazanym przez element ze szczytu stosu i umieszczenie jej na nim. Można by przypuszczać, że uzyskanie aktualnej wartości zmiennej BASE nastąpi poprzez BASE @ . , jednak na ekranie zawsze pojawiać się będzie 10.

## FORMATOWANIE WYDRUKÓW

Jedną z zalet FORTH-a jest możliwość wyprowadzenia liczb w dokładnie zadanej postaci. Dzieje się to przy użyciu następujących słów:

.R — dla liczb pojedynczej precyzji ze znakiem

D.R — dla liczb podwójnej precyzji ze znakiem

U.R. — dla liczb podwójnej precyzji bez znaku

powodują one wyświetlenie drugiej liczby ze stosu na polu o szerokości określonej przez pierwszy element ze stosu. Liczba jest wyświetlana od prawej krawędzi pola. Jest to szczególnie wygodne przy drukowaniu zestawień dużych ilości liczb w taki sposób, że ich najmniej znaczące cyfry występują jedna pod drugą.

Dla wygody podamy omówione dotychczas słowa i sekwencje FORTH-a, przy czym PNS oznaczać będzie Pierwszy Na Stosie, 2 NS — Drugi Na Stosie itd.:

. — powoduje wyświetlenie PNS jako liczby pojedynczej precyzji ze znakiem, niszczy PNS

U. — podobnie jak ., lecz traktuje liczbę Jako zapisaną bez znaku

D. — powoduje wyświetlenie PNS, 2NS jako liczby podwójnej precyzji ze znakiem

N BASE! — wymusza pracę w systemie o podstawie n HEX — wymusza pracę w systemie szesnastkowym DECIMAL — wymusza pracę w systemie dziesiętnym

BASE @ — wyprowadza aktualną wartość zmiennej BASE

.R — niszczy PNS, 2NS wyświetla 2NS jako liczbę pojedynczej precyzji ze znakiem, na polu o szerokości określonej przez PNS, od jego prawej krawędzi

U.R — jako .R lecz traktuje PNS, 2NS jako liczbę bez znaku

D.R — niszczy PNS, 2NS, 3NS, wyświetla 2NS, 3NS jako liczbę podwójnej precyzji ze znakiem, na polu o szerokości określonej przez PNS, od jego prawej krawędzi.

*Dyziek*


http://www.t2e.pl/reduksy

# BASIC XL

W ciągu paru lat firma Atari wypuściła na rynek kilka modeli komputerów 8-bitowvch różniących się wielkością pamięci RAM.

Pierwsze modele nie miały na stałe wbudowanego interpretera żadnego języka, przy zakupie dołączany był cartridge z BASIC-em. Atari zdecydowała się na pełną kompatybilność swoich produktów 8-bitowych i w zasadzie do dziś jest w tym konsekwentna. Dołączany na cartridge'ach BASIC został wbudowany na stałe i tak powstały modele XL. Atari BASIC zajmuje 8 KBajtów pamięci i stanowi najsłabszy punkt komputerów firmy Atari. Próbowano zaimplementować standard Microsoft, ale nie dało to spodziewanych efektów, nadal BASIC pozostawiał wiele do życzenia. Dopiero w 1980 roku amerykańska firma Optimized Systems Software wypuściła produkt, który w znaczący sposób zmienił możliwość i łatwość obsługi komputerów Atari. Produktem tym jest interpreter o nazwie BASIC XL. Wiadomym |est, że im lepszy interpreter, tym więcej potrzebuje pamięci, a firma OSS postawiła sobie za cel, że nowy BASIC będzie zajmował obszar adresowy dotychczasowego BASIC-a (tzn. 8 KB) i że będzie z nim w pełni kompatybilny. Interpreter XL fizycznie zajmuje 16 KBajtów pamięci, które podzielone są na bloki po 4 KBajty. a sprzętowo przy użyciu szybkich układów TTL do obszaru adresowego dołączone są zawsze dwa bloki (aktualnie wykorzystywane). W ten sposób z 16 KB komputer „widzi" tylko 8 KB. Z tego powodu interpreter ten musi być wbudowany na stałe do komputera, a nie można go zapisać na dysku czy kasecie tak jak interpretery czy kompilatory innych języków.

Na czym polega wyższość BASIC-a XL nad Atari BASIC?

Przede wszystkim na szybkości działania, BASIC XL działa 4—6 razy szybciej, również pisanie programów jest szybsze, łatwiejsze i efektywniejsze dzięki dodatkowym instrukcjom.

Charakterystykę BASIC-a XL można ująć w 3 punktach:

1. **Szybkość działania:** jak już wspomniano Atari BASIC i BASIC XL są ze sobą kompatybilne tylko w jedną stronę, tzn. programy napisane w Atari BASIC będą działały „pod" BASIC-em XL poprawnie (pod pewnymi warunkami, o których na końcu artykułu), lecz ze zwiększoną prędkością, co trzeba brać pod uwagę np. w pętlach czasowych. Oczywiście programy napisane w BASIC-u XL przy użyciu wyłącznie instrukcji dostępnych w Atari BASIC będą działały poprawnie w tym ostatnim, ale takie wykorzystanie BASIC-a XL mija się z celem. W BASIC-u XL istnieje instrukcja FAST znacznie przyspieszająca realizację programu. Działanie jej polega na zapamiętaniu fizycznych adresów pamięci, w których zapisane są konkretne linie BASIC-a, do których istnieją odwołania instrukqami GOTO, GOSUB, ON...GOTO. ON...GOSUB oraz linie, w których realizowane są pętle FOR/N EXT.

2. **Szybkość i łatwość pisania programów:** BASIC XL daje możliwość korzystania z automatycznego numerowania linii programu od dowolnego numeru z dowolnym krokiem (numeracja nie może przekroczyć 32767). Służy do tego instrukcja NUM. Często zachodzi potrzeba przenumerowania programu i można wtedy skorzystać z instrukcji RENUM. Przenumerowanie trwa kilka sekund np. dla programu o długości 1300 linii (ok. 16 KB) niecałe 3 sekundy. Korzystając z instrukcji RENUM można podać dowolny początkowy numer linii i krok, jeśli nie podamy parametrów, to interpreter sam założy te wielkości. W odróżnieniu od Atari BASIC, BASIC XL „przyjmuje" instrukcje napisane w dowolny (byle poprawny) sposób. Można pisać dużymi i małymi literami, inverse video jest też dozwolone. Z tego względu, aby uzyskać napisy w inverse video (w negatywie) lub małymi literami tekst należy koniecznie umieścić w cudzysłowie (jest to szczególnie ważne przy napisach umieszczonych w liniach DATA). Komputer podczas listowania programu wyświetli tekst w jednolitej formie niezależnie od tego, jak został do niego wprowadzony. Zapisany on jest małymi literami, tylko pierwsze litery instrukcji, nazw zmiennych i tablic są duże. Dodatkowo każda pętla FOR/NEXT oraz IF/ELSE. WHILE/ENDWHILE jest przesunięta o kilka spacji. Na skutek tych zabiegów listowany tekst programu jest bardziej czytelny. Czasami zdarza się, że przy uruchamianiu programów wystąpi błąd i wówczas BASIC XL podaje nie tylko numer błędu, ale również jego interpretację w formie opisu. Bardzo łatwo można określić numer błędu i linię, w której wystąpił przy użyciu instrukcji ERR (istotne np. przy programowej obsłudze błędów). Wprowadzona instrukcja SET pozwala w prosty sposób „sterować" komputerem i możliwościami samego interpretera np. sterować klawiszem BREAK, tabulacją, rodzajem znaków uzyskiwanych instrukcją INPUT, wreszcie można ustawić interpreter na całkowitą kompatybilność z Atari BASIC, lecz wtedy BASIC XL będzie pracował bez wyżej wspomnianych funkcji. Aktualne wartości parametrów instrukcji SET można odczytać rozkazem SYS. Czasami pisząc długie programy mamy problem czy nazwa jakiejś zmiennej nie występowała już w programie, aby to sprawdzić można użyć instrukcji LVAR, która poda nam nie tylko wszystkie nazwy zmiennych występujących w programie, ale również numery linii, w których występują. Dla ułatwienia i przyspieszenia pracy z dyskami w BASIC-u XL zawartych jest wiele instrukcji normalnie dostępnych z DOS-u, takich jak DIR (spis zawartości dysku), ERASE (usuwanie plików), PROTECT i UNPROTECT (zabezpieczanie i odbezpieczanie zbiorów), RENAME (zmiana nazwy zbioru) i inne. Dla początkujących programistów oraz przy kłopotach z uruchomieniem POPRAWNIE działającego programu przydatna będzie instrukcja TRACE czyli śledzenia programu. Podaje ona numery aktualnie wykonywanych linii programu i pozwala dzięki temu prześledzić jego działanie.

3 **Efektywność:** BASIC XL posiada o 45 instrukcji więcej niż Atari BASIC, wiele z nich pozwala na proste formułowanie warunków logicznych, uzyskanie informacji o dostępnej pamięci, zmianę zawartości poszczególnych komórek (całych obszarów) pamięci. Specjalne instrukcje z Player/Missile Graphics w prosty i łatwy sposób pozwalają tworzyć ruchome obrazy do gier. W BASIC-u XL nowego znaczenia nabrała deklaracja DIM. Możliwe są do zadeklarowania tablice tekstowe wieloelementowe, z których każdy jest osobnym ciągiem znaków. Np. DIM A$(4,40) oznacza tablicę czteroelementową, w której każdy element może być ciągiem tekstowym o długości 40 znaków. W BASIC-u XL można wykonywać operacje logiczne na bitach. Służą do tego operatory &. !, % — AND, OR, EOR. Instrukcją DEL możemy szybko kasować niepotrzebne linie programu (jedną instrukcją DEL można skasować dowolną liczbę linii). Często w programach zachodzi potrzeba „zarezerwowania" pewnego obszaru pamięci np na program ANTIC-a, czy jakąś procedurę maszynową, pomocną wtedy staje się instrukcja LOMEN. W BASIC-u XL rozbudowano instrukcję warunkową IF... THEN o instrukcję IF...ELSĘ...ENDIF. Bardzo wygodną instrukcją jest MOVE. Pozwala ona szybko i skutecznie „przenieść" dowolny fragment pamięci w inne miejsce (oczywiście tylko w obszarze adresowanym komputera) Należy być ostrożnym z tą instrukcją, gdyż przy jej użyciu nie są sprawdzane żadne adresy i można przez nieuwagę „zawiesić" komputer. W BASIC-u XL istnieje oprócz instrukcji IF. .THEN oraz IF...ELSE pętla warunkowa WHILE/ENDWHILE. Pętla ta wykonywana jest tak długo, dopóki zmienna sterująca pętli ma wartość różną od zera. Sprawdzanie wartości zmiennej sterującej odbywa się na początku pętli. Do szybkiego prostego przesyłania do i z komputera danych, np. danych o obrazie, można wykorzystać instrukcje BGET, BPUT. Pozwalają one przesyłać dowolną liczbę bajtów do lub z komputera poprzez kanały transmisji szeregowej, przy czym w obydwu instrukcjach określa się liczbę bajtów do przesłania oraz adres pamięci, od którego dane mają być transmitowane. Do BASIC-a XL wprowadzono tzw. dynamiczną instrukcję INPUT, działa ona tak jak analogiczna instrukcja BASIC-a w Spectrum czy C-64. Nie trzeba, przy wczytywaniu tekstu, deklarować tablicy, gdyż instrukcja ta automatycznie deklaruje tablicę o podanej nazwie i wymiarze 40 elementów. Dosyć istotnym udogodnieniem jest instrukcja PRINT USING pozwalająca w prosty sposób formatować wydruki. Jeżeli określamy format wydruku liczb, to mogą być one drukowane ze znakiem lub bez, miejsca nieznaczące mogą być wypełnione spacjami, zerami bądź gwiazdkami, liczby z przecinkiem dziesiętnym drukowane są w zaokrągleniu do tylu miejsc po przecinku, ile przewiduje format wydruku. Wydruk jest wyrównywany do prawej strony. W przypadku tekstu można wyrównywać wydruk do prawej bądź lewej strony. Do przesyłania całych rekordów i ciągów (tablic) do i z komputera służą instrukcje RGET i RPUT. Instrukcją TAB można przesłać dowolną liczbę spacji do dowolnego urządzenia (przydatne np. przy centrowaniu wydruku na drukarce). Często zachodzi konieczność korzystania z generatora liczb losowych, lecz zakres liczb powinien być większy niż (0,1). Używając instrukcji RANDOM można uzyskać liczbę losową z zakresu od zera do podanej liczby, bądź z podanego przedziału liczb. Przy pracach z różnymi tekstami zachodzi często potrzeba odnalezienia pewnego tekstu, bądź sprawdzenia czy poszukiwany tekst występuje w innym tekście. Służy do tego instrukcja FIND, która podaje miejsce występowania poszukiwanego ciągu znaków w ciągu przeszukiwanym. Podobne znaczenie dla programistów mają instrukcje LEFTS, RIGHT$ i MIDS, które tworzą z ciągu wzorcowego subciągi o określonej ilości znaków odpowiednio od lewej i prawej strony oraz od-do znaku. Przy obsłudze portów joysticków pomocne są instrukcje: HSTICK — podaje poziome wychylenie joysticka oraz VSTICK — podaje pionowe wychylenia joysticka. Dodatkowo funkcja PEN podaje pionowe i poziome położenie pióra świetlnego.

Przy pisaniu gier bardzo pomocne są specjalne instrukcje Player/Missile Graphics:

BUMP — podaje zawartość rejestrów kolizji.

PMADR — podaje położenie w pamięci obszarów, w których przechowywane są wzory graczy lub pocisków.

DPEEK i DPOKE — pozwalają odczytać lub zapisać dwie sąsiednie komórki pamięci. W Atari BASIC odpowiednikiem DPEEK jest sekwencja PEEK (adr) + 256 * PEEK (adr + 1), gdzie „adr" to konkretny adres. Podczas pisania programów maszynowych często adresy podawane są w zapisie szesnastkowym, chcąc łatwo zamienić dowolną liczbę dziesiętną na szesnastkową (hexadecymalną) wystarczy użyć funkcji HEXS. W trybie bezpośrednim można bez kłopotu zamienić liczbę hex na dziesiętną, wystarczy przed liczbą hex napisać znak dolara ($).

PMCLR — instrukcja ta „czyści" obszar, w którym zapisany jest obraz gracza lub pocisku.

PMCOLOR — analogicznie jak instrukcja SETCOLOR. lecz dla rejestrów 704-707.

PMGRAPHICS — zezwala bądź blokuje grafikę PMG oraz pozwala wybrać grafikę o pojedynczej lub podwójnej rozdzielczości.

PMMOVE — pozwala przesuwać obraz gracza lub pocisku na ekranie.

PMWIDTH — pozwala wybrać szerokość dowolnego gracza (pojedynczą, podwójną lub poczwórną).

MISSILE — w prosty sposób pozwala uzyskać efekt „oddania" strzału przez gracza.

Na koniec jeszcze małe wyjaśnienia odnośnie kompatybilności. W przypadku programów zapisanych na taśmę lub dysk instrukcjami CSAVE lub SAVE i odczytywanymi przez CLOAD lub LOAD nie powinno być żadnych problemów i niespodzianek. Przy korzystaniu z instrukcji LIST i ENTER mogą się pojawić kłopoty. Z czego one wynikają? Otóż, jak zapewne każdemu wiadomo, nazwy zmiennych i tablic nie mogą rozpoczynać się od słów kluczowych Przykładowo w programie napisanym w Atari BASIC użyliśmy sekwencji NUMBER = 7. Po wczytaniu programu pod BASIC-em XL, interpreter zidentyfikuje taką nazwę jako NUM BER = 7. czyli jako instrukcję NUM oraz zmienną BER, która, jeżeli nie była wcześniej użyta w programie, to nie będzie równa 7 i w konsekwencji wykonana będzie instrukcja NUM 0 (automatyczne numerowanie linii od numeru 0). Aby program działał poprawnie należy użyć instrukcji LET NUMBER = 7 Programy zapisane w BASIC-u XL me będą działały w Atari BASIC, po wczytaniu instrukcją ENTER. gdyż jak wiadomo Atari BASIC nie przyjmuje instrukcji napisanych małymi literami lub w inverse video

Jeżeli ktoś uważa, że jego programy pracują zbyt wolno, to można skompilować program napisany w BASIC-u XL kompilatorem o nazwie TURBO BASIC XL COMPILER. Należy przy tym uważać na niektóre instrukcje, gdyż wspomniany kompilator został napisany dla potrzeb użytkowników TURBO BASIC-a XL. Po prostu w TURBO BASIC-u XL użyto innych tokenów dla kilku instrukcji występujących w BASIC-u XL. Skompilowane programy działają około 10 razy szybciej niż programy napisane w Atari BASIC.

Podsumowując należy podkreślić, że opisany produkt firmy OSS powinien zadowolić każdego programistę dzięki swojej prostocie, szybkości działania i prostej obsłudze błędów. Pewnym mankamentem może być brak kilku instrukcji dostępnych w TURBO BASIC-u XL np. TEXT, TIME$, CIRCLE, FILL, ale co stoi na przeszkodzie, aby stworzyć odpowiednie procedury maszynowe (przykład można znaleźć w Bajtku nr 1/87).

Po rocznej eksploatacji BASIC-a XL doszedłem do wniosku, że ci, którzy raz spróbują napisać coś w tym języku nieprędko wrócą do Atari BASIC.

W Polsce BASIC XL można wbudować do komputera w serwisie Atari. Proponowana cena jest, moim zdaniem, przystępna jak na taki produkt, aczkolwiek stanowi pewne obciążenie budżetu.

*Arrow*


http://www.t2e.pl/reduksy

# GENERATOR RYTMÓW

**Przypuszczam, że wielu początkujących muzyków czy też muzykantów ma pewne kłopoty z utrzymaniem rytmu. Sądzę tak, ponieważ sam jestem (i zapewne już zostanę) takim właśnie muzykantem. A do tego dźwięk przedstawionej tu „elektronicznej perkusji” jest ciekawszy, bardziej inspirujący niż tupanie nogą, czy suchy stukot metronomu.**

Dla zapisu rytmów dysponujemy polem o szerokości 39 kolumn i wysokości 2 wierszy. Początek pola wskazuje znak „?”. Tam też pojawia się kursor. Każdy wiersz określa rytm jednego bębenka. Górny wiersz — małego, dolny — większego. Cyfry widoczne nad polem rytmów to numery kolumn. Numery są dwucyfrowe — czytamy je z góry w dół. Poszczególne kolumny definiują kolejne elementarne jednostki czasu, w których może zabrzmieć dźwięk. Programowanie generatora polega na umieszczeniu w polu rytmów znaczków wskazujących, kiedy dany bębenek ma grać. Mogą być użyte dowolne znaki z wyjątkiem „Q”, „ ” i „I”. Spacja definiuje bowiem ciszę, pauzę czyli brak dźwięku, pionowa kreska zaś (klawisze <SHIFT> i < = >) użyta w zapisie górnego bębenka oznacza koniec cyklu, to znaczy, że jako następna będzie interpretowana kolumna nr 1. Jeśli nie użyjemy takiej kreski, to cykl będzie trwał całe 39 jednostek. Po wpisaniu rytmu naciskamy RETURN i perkusja gra. Powtórne naciśnięcie RETURN — i cisza. Można poprawić zapis, itd. Aby opuścić program należy tuż za pytajnikiem wprowadzić <Q> (quit). Zwracam uwagę, że każdy kulturalny program powinien kończyć się w kontrolowany sposób, a nie tylko przez BREAK czy też wyciągnięcie wtyczki z kontaktu.

Kilka słów na temat programu. To mocna strona komputerów ATARI, że z ich nieprzebranego bogactwa dźwięków można korzystać tak prosto! Tu konwencjonalna pętelka cichnącego szumu w liniach 260..290 doskonale spełnia swe zadanie. Szumy są dwa: cieńszy (linia 270) i grubszy (linia 280). Głośność dźwięku w każdym momencie jest mnożona przez 0, jeśli kolejne miejsce na ekranie jest puste, zaś przez 1 w przeciwnym wypadku Wyrażenie logiczne bowiem, a takim jest np. (PEEK (EKR + I)>0), przyjmuje zgodnie z konwencją Atari BASIC-a jedną z dwu wartości: 1 (prawda) lub 0 (fałsz). Prostota i szybkość tego sposobu zasadza się na bezpośrednim spoglądaniu w pamięć ekranu. Zmienna CS ma tu znaczenie uboczne, a sama instrukcja INPUT służy tylko do zatrzymania programu, gdy trzeba dokonać zmian. Przed podjęciem nowego rytmu bada w liniach 210..230, czy wprowadzono ograniczenie długości cyklu i stosownie do tego ustala N.

Wyjaśnienie niektórych PEEK'ów i POKE'ów. Komórka 82 (linie 100 i 300) ustala lewy margines ekranu. W komórkach 88 i 89 (linia 130) znajduje się dwubajtowy adres pamięci ekranu. Zmienna EKR będzie więc wskazywać na nasze pole rytmów Pod adresem 752 można (w linii 180) zgodzić się, by kursor przebywał na ekranie, lub (w następnej) zabronić mu tego. W komórce 764 (linie 170, 240) jest kod ostatnio naciśniętego klawisza, lub — gdy niczego się nie naciska — liczba 255.

*Janusz B.Wiśniewski*

```
100 GRAPHICS 0:DIM C$(1):POKE 82,1
110 POSITION 4,0:? " Generator";
120 ? " rytmow    (c) JBW '87 ";
130 EKR=PEEK(88)+PEEK(89)*256+160
140 POKE 566,158:FOR I=1 TO 39
150 J=INT(I/10):POKE EKR-80+I,J+16
160 POKE EKR-40+I,I-10*J+16:NEXT I
170 POKE 764,255
180 POKE 752,0:POSITION 0,4:INPUT C$
190 POKE 752,1:POSITION 0,4:PRINT " "
200 IF C$="Q" THEN 300
210 FOR N=1 TO 39
220 IF PEEK(EKR+N)=124 THEN 240
230 NEXT N
240 IF PEEK(764)<255 THEN 170
250 FOR I=1 TO N-1
260 FOR J=15 TO 0 STEP -5
270 SOUND 0,0,0,J*(PEEK(EKR+I)>0)
280 SOUND 1,17,0,J*(PEEK(EKR+40+I)>0)
290 NEXT J:NEXT I:GOTO 240
300 POKE 82,2:POKE 752,0:POKE 566,1
```

# TAJEMNICE ATARI (3)

**Ten odcinek „Tajemnic Atari” będzie inny — poświecony tylko jednej grze.**

**MINDSHADOW — to jedna z ciekawszych gier tekstowo--obrazkowych.**

Większość „użytkowników” komputerów ogarnięta jest panicznym strachem przed językiem angielskim, dlatego może rezygnuje z gier tego rodzaju. Tymczasem są one bardzo interesujące i mogą być doskonałą rozrywką, a przy okazji pozwalają doskonalić jakże przydatną znajomość języków obcych. Aby pomóc bojaźliwym niedowiarkom i przekonać ich do gier tekstowo-obrazkowych, podamy wskazówki jak uczynić pierwszy krok wśród cieni świadomości (Mindshadow).

Obudziłeś się na opuszczonej plaży ogarnięty amnezją. KIM JESTEŚ!? Nie wiesz. Nie potrafisz odpowiedzieć na żadne, nasuwające Ci się, pytanie. Próbujesz chodzić w różnych kierunkach — wydostać się z wyspy. Szukasz miejsca, w którym znajdziesz odpowiedzi na dręczące Cię pytania...

Idąc na północ N dotarłeś do prymitywnej chaty, wchodzisz do niej ENTER HUT, rozglądasz się i bierzesz garść słomy GET STRAW. Opuszczasz chatę i wyruszasz w dalszą drogę S — E , znalazłeś starą, rozbitą łódź. Musisz zabrać kawałek stalowej konstrukcji jej szkieletu GET STEEL i skierować się na wschód E. Znalazłeś się w dżungli pełnej winorośli i paproci, ścinasz długi, mocny, winny pęd GET VINE i wyruszasz w dalszą drogę. Znowu łódź, chata, pusta plaża — docierasz do skalistego wybrzeża W — W — S — E. Musisz położyć wszystkie rzeczy, które niosłeś ze sobą, chwycić winorośl i przywiązać ją mocno do skały

DROP ALL — GET VINE — TIE VINE ON ROCK. Następny krok to zsunąć się w dół DOWN. Znajdziesz się przed wejściem do jaskini. Wejdź do ciemnego, zimnego wnętrza W i rozpocznij kopanie DIG. O szczęśliwcze — znalazłeś mapę, weź ją i obejrzyj GET MAP — READ MAP. Zabierz ze sobą kawałek skały z rumowiska i ruszaj w powrotną drogę GET ROCK — E — UP. Wracając weź pozostawione przed chatą przedmioty GET ALL i udaj się w kierunku pierwszej chaty W — N — N. W ten sposób dotrzesz do ruchomych piasków — nie masz innego wyjścia — musisz spróbować je ominąć błądząc po górach N — E — E — S — S — E. Nareszcie! Jakaś inna plaża, podobnie pusta jak poprzednia, ale znalazłeś butelkę. Spróbuj ją zabrać ze sobą GET BOTTLE i z powrotem w góry — w kierunku ruchomych piasków, chaty i plaży W — N — N — W — W — S — W — S — S — S. Jesteś już u celu swej wędrówki po wyspie. Potrzyj więc kawałkiem stali o kamień, a wzniecisz ogień RUB STEEL WITH-ROCK. Tak! Tak! Udało się! Zostałeś spostrzeżony ze statku przepływającego opodal, ale gdy łódź dobije do brzegu, kapitan — stary wilk morski — powie: „Nie zabieram żadnych pasażerów”. Masz jeszcze tylko jedną szansę, może znaleziona butelka okaże się przepustką do opuszczenia wyspy — oddaj ją kapitanowi GIVE BOTTLE. Jego uśmiech mówi wszystko, masz „bilet” na statek.

W ten sposób znalazłeś się na starym pirackim szkunerze z 1890 roku — udało Ci się opuścić wyspę. Ale to dopiero początek trudnej drogi. Czujesz się teraz jak więzień. Dalej radź sobie sam. Na drogę dostaniesz kilka wskazówek: przerwij łańcuch kotwiczny na kabestanie i dostań się do okrętowej kuchni, choć załoga nie pozwala ci przejść przez południowe drzwi.

Miłej zabawy...

*Sergiusz Piotrowski*
*Tomasz Mazur*




**Ostatni opisywany układ Atari — PIA (Peripherial Interface Adaptor) — to standardowy układ wejścia/wyjścia serii 65xx: 6520. Jest on wykorzystywany do zarządzania pamięcią oraz do obsługi wejść joysticków.**

PIA posiada dwa niezależne, programowalne 8-bitowe porty wejścia/wyjścia. Port A obsługuje wejścia joysticków, a port B służy do sterowania MMU (Memory Management Unit — układ zarządzania pamięcią). Dla każdego z nich przewidziane są dwa rejestry: rejestr portu i rejestr kontroli portu.

Poszczególne bity w rejestrze PORTA odpowiadają następującym funkcjom:

| bil | znaczenie |
|---|---|
| 0 | joystick 1 — naprzód |
| 1 | joystick 1 — wstecz |
| 2 | joystick 1 — lewo |
| 3 | joystick 1 — prawo |
| 4 | joystick 2 — naprzód |
| 5 | joystick 2 — wstecz |
| 6 | joystick 2 — lewo |
| 7 | joystick 2 — prawo |

Normalnie wszystkie bity PORTA są ustawione (1) Skasowanie bitu w tym rejestrze oznacza poruszenie joysticka w odpowiednim kierunku.

Kolejnym bitom w rejestrze PORT B odpowiadają różne obszary pamięci RAM i ROM komputera:

| bit | znaczenie |
|---|---|
| 0 | system operacyjny (1—włączony) |
| 1 | interpreter BASIC-a (O — włączony) |
| 7 | program testujący (O — włączony) |

Bity 2-6 są niewykorzystane w komputerach 800XL i 65XE. W modelu 130XE bit 4 steruje dostępem mikroprocesora 6502 do dodatkowych banków pamięci, a bit 5 — dostępem ANTIC-a Bity 2 i 3 wybierają jeden z czterech dodatkowych banków pamięci. W modelu 256XT bity 2-5 wybierają numer jednego z 16 banków pamięci dostępnego w obszarze adresowanym 16384-32767

Rejestry kontroli PORTACNTL i PORT/BCNTL służą do sterowania portami PIA oraz dodatkowymi liniami sygnałowymi CA1, CA2, CB1 i CB2. Linie te są dołączone do złącza szeregowego i służą do obsługi urządzeń zewnętrznych. Po linii CA1 przesyłane są meldunki urządzeń zewnętrznych, CA2 steruje silnikiem magnetofonu, po CB1 przesyłane są żądania przerwań, a po CB2 rozkazy. Przykładowo wpisanie wartości 54 do PORTACNTL uruchamia silnik magnetofonu, a 60 zatrzymuje go.

Z powyższego opisu wynika, że porty PIA są układami wejścia, a na początku napisałem, że są to porty wejścia/wyjścia. To nie jest pomyłka. Przez wpisanie odpowiednich wartości do rejestrów kontroli można ustawić porty jako układy wyjścia. To niepozorne sformułowanie kryje w sobie ogromne możliwości: gdy port A pracuje jako wyjście, to z gniazd joysticków otrzymujemy 8-bitowy sygnał w standardzie TTL. Można go bezpośrednio użyć do sterowania 8 urządzeń zewnętrznych (np. manipulatorów), a po zastosowaniu dekodera (17 demulti- plekserów UCY 74154) do sterowania 256 różnych urządzeń! Daje to ogromne możliwości zastosowania komputera jako inteligentnego sterownika. Temat ten jeszcze poruszymy w przyszłości.

Teraz pora na małą porcję adresów:

| | | |
|---|---|---|
| 54016 | PORTA | portA |
| 54017 | PORTB | port B |
| 54018 | PORTACNTL | rejestr kontroli portu A |
| 54019 | PORTBCNTL | rejestr kontroli portu B |
| 632-633 | JOY0-JOY1 | położenie joysticków |

*Wojciech Zientara*


http://www.t2e.pl/reduksy

# KLAN ATARI

# BAZY DANYCH

**Różnego typu bazy danych są najczęściej używanym rodzajem programów użytkowych na wszystkie komputery domowe. Często są to programy pisane przez samych użytkowników — wynika to z faktu, że baza danych jest efektywna przy pracy ze stacją dysków, nie ma więc takich programów dla użytkowników magnetofonów. Poniżej scharakteryzowane są najpopularniejsze programy tego typu dostępne na Atari.**

**SynFile+**
(1983 — Synapsę Software)

Najlepsza aktualnie dostępna baza danych. Kształt rekordu jest tworzony przez użytkownika i jest ograniczony jedynie powierzchnią ekranu — 20 wierszy po 80 znaków. W rekordzie mogą się zna|dować pola tekstowe, liczbowe (różnych typów), tablicowe i warunkowe. Zawartości niektórych pól mogą być obliczane przez program Dostępnych jest wiele różnorodnych funkcji, takich jak kopiowanie plików, ich łączenie, tworzenie podzbiorów, przeszukiwanie wg dowolnych parametrów oraz tworzenie list i etykiet z dowolnych elementów plików. Ważną zaletą jest możliwość korzystania z plików SynFile'a przez inne programy: SynCalc, SynStat, SynGraph. AtariWriter, Paper Clip i wiele innych. Użytkowanie programu jest bardzo proste dzięki rozwijającemu się w dolnej części ekranu menu z dostępnymi w danej opcji funkcjami.

**Home Filing Manager**
(1982 — Atari Inc.)

Jedna z pierwszych baz danych na Atari. Jest zbudowana w formie notesu — dane zapisuje się w dowolnym formacie w polu o wielkości ekranu (22 wiersze po 40 znaków), jakby na kartce papieru. Program nie jest wygodny z powodu małej ilości funkcji. Ponadto dane są zapisywane na dyskietce w niestandardowym formacie Cały program bardziej niż bazę danych przypomina podręczny notatnik i jako taki powinien być wykorzystywany.

**GT DataManager**
(1983 — ELCOMP Publishing)

Stosunkowo prosta baza danych, wygodna głównie w przypadku niewielkich plików. Kształt rekordu definiowany przez użytkownika może się składać z ośmiu pól o długości do 37 znaków każde. Wadą jest niewielka ilość funkcji, dodatkowo korzystanie z programu utrudniają nieprzejrzyste ich sformułowania. Może być polecana jedynie do katalogowania stosunkowo krótkich zbiorów, w których występuje niewiele informacji.

*Wojciech Zientara*

# KLAN SPECTRUM

# OBRACANIE ZNAKÓW

**Napisanie dłuższego programu w języku maszynowym jest bardzo żmudne i pracochłonne, a często wręcz nieopłacalne. Warto jednak łączyć procedury maszynowe z programem w BASIC-u. Wiele pism publikuje krótkie procedury przeznaczone do dołączania do własnych programów.**

Prezentowana poniżej procedura pozwala rozszerzyć możliwości graficzne ZX-Spectrum. Umożliwia ona obrócenie dowolnego znaku graficznego na ekranie. Obrócone litery są bardzo przydatne na przykład do opisywania tabel lub wykresów. Dzięki niej unikniemy konieczności definiowania całego, nowego generatora znaków.

Procedura jest umieszczana w pamięci RAM za pomocą następującego programu w BASIC-u.

```
10 CLEAR 65299: LET a=65299
20 FOR k=100 TO 105
30 LET s=0
40 FOR l=1 TO 10
50 LET a=a+1: READ b: POKE a,b: LET s=s+b: NEXT l
60 READ m: IF m<>s THEN PRINT "BLAD W ";k
70 NEXT k
100 DATA 58,176,92,254,24,210,134,12,205,158,1323
101 DATA 14,58,177,92,254,32,210,134,12,133,1116
102 DATA 111,229,17,80,255,6,8,126,18,19,869
103 DATA 36,16,250,225,14,8,221,33,80,255,1138
104 DATA 6,8,221,203,0,46,203,22,221,35,965
105 DATA 16,246,36,13,32,236,201,201,201,0,1182
```

Następnie należy nagrać ją na magnetofon przy pomocy rozkazu SAVE „rotate" CODE 65300, 60. Program, w którym będziemy |ą wykorzystywać, musi zawierać instrukcje wczytujące dane uprzednio zapisane na taśmie: CLEAR 65299 : LOAD „rotate" CODE.

W jaki sposób należy posługiwać się procedurą w programie? Najpierw drukujemy odpowiedni znak na ekranie i wpisujemy jego współrzędne do komórek NOT-USED w obszarze zmiennych systemowych. Służą do tego instrukcje POKE Ź3728.W : POKE 23729,K (gdzie: W — numer wiersza, K — numer kolumny). Następnie wywołujemy procedurę przy użyciu funkcji RANDOMIZE USR 65300. Jeżeli instrukcje te realizowane są bezpośrednio jedna po drugiej, to działanie procedury jest tak szybkie, że jesteśmy w stanie zaobserwować na ekranie tylko obrócony znak.

Króciutki program zilustruje działanie procedury:

```
5 INPUT "PODAJ NUMER KOLUMNY 0-31";M
10 LET a$="TEST PROGRAMU OBROT "
20 POKE 23729,M
30 FOR K=0 TO 21
40 POKE 23728,K
50 PRINT AT K,M;A$(22-K)
60 RANDOMIZE USR 65300
70 NEXT K
```

Dociekliwy czytelnik na pewno spróbuje uruchomić procedurę po podaniu parametrów przekraczających dopuszczalne współrzędne ekranu. Nic z tego! Kończy się to przerwaniem działania całego programu i napisaniem w oknie systemowym komunikatu błędu OUT OF SCREEN.

Załączamy również listing procedury w asemblerze. Rozpoczyna się ona od załadowania do rejestru A numeru wiersza (linia 140) i sprawdzenia, czy nie jest on większy od dopuszczalnego. W przypadku przekroczenia dozwolonej współrzędnej następuje skok pod adres # 0C86 (OUT OF SCREEN). Jeżeli numer wiersza podano prawidłowo, to wywoływana jest procedura obliczająca jego adres w pamięci obrazu. Adres ten znajduje się w parze rejestrów HL. Następnie do rejestru A ładowany jest numer kolumny i po sprawdzeniu jego prawidłowości dodawany do adresu wiersza (linie 180 — 220). Para rejestrów HL zawiera teraz adres obracanego znaku na ekranie. Pętla w liniach 240 do 300 przesyła osiem bajtów tworzących znak do bufora zaczynającego się od adresu 65360. Dalsza część procedury to właściwy obrót znaku. Polega to na arytmetycznym przesuwaniu kolejnych bajtów bufora w prawo, podczas którego bit przeniesienia (Carry) jest ustawiany lub zerowany w zależności od wartości bitu O (linia 350). Carry jest podstawą do utworzenia nowej wartości bajtu na ekranie (linia 360). Poprzez umieszczenie w liniach 350 i 360 różnych kombinacji rozkazów SRA (IX), SLA (IX), RL (HL), RR (HL) uzyskujemy obrót w lewo, obrót w prawo, odbicie lustrzane z obrotem.

*Janusz Jarmoch*

```
90 ;*********************************************
91 ;
92 ; OBROT ZNAKU GRAFICZNEGO
93 ;
94 ;*********************************************
100 ORG 65300
110 NRWR    EQU     23723    ;  NUMER WIERSZA
120 NRKL    EQU     23729    ;  NUMER KOLUMNY
130 BUFOR   EQU     65360
140         LD      A,(NRWR)
150         CP      24
160         JP      NC,#0CS6 ;  OUT OF SCREEN
170         CALL    #0E9E    ;  ZNAJDZ ADRES POCZ. WIERSZA
130         LD      A,(NRKL) ;  W PAMIECI OBRAZU
190         CP 32
200         JPM     C,#0C86
210         ADD     A,L      ;  DODAJ NR. KOLUMNY DO ADR.
220         LD      L,A      ;  WIERSZA
230         PUSH    HL       ;  ZAPAMIETAJ ADR. ZNAKU
240         LD      DE,BUFOR ;  DE-ADRES BUFORA
250         LD      B,8      ;  PRZESLIJ 8 BAJTOW Z
260 PTLBF   LD      A,(HL)   ;  PAMIECI OBRAZU DO BUFORA
270         LD      (DE),A
280         INC     DE
290         INC     H
300 DJNZ PTLBF
310         POP     HL
320         LD      C,8
330 NASTL   LD      IX,BUFOR ;  PETLA DOKONUJACA OBROTU
340         LD B,8           ;  ZNAKU GRAFICZNEGO
350 NASTB   SRA     (IX)     ;  PETLA TWORZACA
360         RL      HL)      ;  NA EKRANIE JEDEN
370         INC IX           ;  JEDEN BAJT
380 DJNZ    NASTB            ;  OBROCONEGO ZNAKU
390         INC     H
400         DEC     C
410         JR      NZ,NASTL ;  TWORZ NASTEPNY BAJT
420         RET              ;  POWROT Z PROCEDURY
```


http://www.t2e.pl/reduksy

# BEZ WYBORU(2)

**Hisoft Pascal jest nie tylko jedynym kompilatorem Pascala dostępnym na komputerze Spectrum. Jest to także jedyny znany mi kompilator, który nie wymaga dyskowej pamięci zewnętrznej. Wynikają z tego pewne zmiany w samym jęxyku, które z kolei pociągaja za soba niemożność efektywnego korzystania z plików dyskowych tym, którzy stacje dysków posiadają, np. użytkownikom Amstradów 6128.**

## PASCAL

Dialekt przyjęty przez firmę Hisoft różni się od standardu opublikowanego przez Niklausa Wirtha.

Po pierwsze założono, że użytkownik posługuje się magnetofonem, a nie stacją dysków. Język pozbawiony został zatem całego systemu plików. Wszystkie operacje WRITE odnoszą się do ekranu, natomiast READ — do klawiatury komputera. Wydruk na drukarce można uzyskać poprzez podanie w procedurze WRITE odpowiedniego kodu sterującego — patrz poniżej. Zamiast plików wprowadzono natomiast możliwość zapisu zawartości zmiennych na taśmie.

Po drugie nie ma możliwości definiowania rekordów z wariantami.

Po trzecie procedury i funkcje nie mogą być parametrami innych procedur i funkcji.

Po czwarte, wprowadzono szereg standardowych procedur i funkcji, które rozszerzają możliwości języka w konkretnych implementacjach.

Oto szczegółowe wymienienie tych odstępstw.

**Nazwy**. Identyfikatory wprowadzane do programu przez programistę są rozróżniane na podstawie pierwszych 10 znaków. Rozróżniane są duże i małe litery.

**Słowa kluczowe**. Wszystkie słowa kluczowe Pascala, jak też nazwy procedur standardowych, typów itp. MUSZĄ być w programie pisane wielkimi literami.

**Liczby**. Wolno używać liczb całkowitych w zapisie szesnastkowym, poprzedzonych w zapisie znakiem #, np. #AF12.

**Stałe**. W definicji stałych dopuszczono możliwość użycia funkcji CHR. Dozwolona jest zatem definicja

```
const   CR = CHR (13);
        BackSpace = CHR (8);
```

Jak widać, konstrukcja taka umożliwia reprezentację znaków sterujących jako stałych znakowych. Poza tym długość stałych tekstowych ograniczona jest do 255 znaków.

**Typy**. Deklarowany przez użytkownika typ wyliczeniowy nie może zawierać więcej niż 255 elementów. Dopuszczalne jest użycie słowa PACKED w odniesieniu do tablic, jakkolwiek nie wywołuje to żadnych skutków. Niektóre typy tablicowe (np. tablice znaków) są zawsze przechowywane w pamięci spakowanej.

Zbiory nie mogą zawierać więcej niż 255 elementów, W definicjach typów wskaźnikowych obowiązuje zasada predefinicji, tzn. najpierw definiujemy typ rekordowy, a następnie wskaźnik do niego. Tablice wielowymiarowe odpowiadają tablicom tablic. Z tego powodu jest obojętne, czy do elementu tablicy odwołujemy się przy pomocy napisu A [1,1] czy A [1]

**Instrukcje**. Nie jest dozwolone użycie CASE jako instrukcji pustej, tzn. na przykład w formie

CASE TRUE OF END;

Dozwolone jest użycie ELSE wewnątrz instrukcji

W przypadku, gdy nie zostaje wykonana żadna z instrukcji wewnątrz CASE, efekt wykonania CASE jest taki, jak instrukcji pustej.

W instrukcji FOR zmienną indeksującą nie może być parametr procedury.

Skoki przy pomocy GOTO są dozwolone w obrębie tej samej jednostki programowej (procedury, funkcji, programu). Nie jest dozwolone wyskakiwanie na zewnątrz pętli FOR. Etykiety, do których następują skoki, muszą być zadeklarowane w tej samej jednostce, w której są używane.

Nagłówek programu. Jest obowiązkowy, chociaż nie pełni żadnej funkcji — wobec braku systemu plików nie zawiera bowiem parametrów. Na przykład:

PROGRAM Euklides;

## OBIEKTY STANDARDOWE

**Stałe**. Wprowadzono stałą standardową MAXINT = 32767, równą największej reprezentowanej dodatniej liczbie typu INTEGER.

**Procedury**. Procedury obsługi wejścia (READ) i wyjścia (WRITE) różnią się od innych dialektów Pascala drobnymi szczegółami. Poza tym Hisoft Pascal wzbogacono o cały szereg procedur standardowych nie figurujących w raporcie języka, a ułatwiających posługiwanie się Pascalem na komputerze 8-bitowym z pamięciową kasetową. Oto one w porządku alfabetycznym.

**ADDR (VAR Z : DowolnyTyp) : INTEGER**

Parametrem funkcji jest zmienna dowolnego typu, wynikiem — adres tej zmiennej w pamięci. Należy nadmienić, że adres ten jest reprezentowany jako INTEGER. Z tego powodu adresy większe od 32767 są przedstawiane w postaci liczb ujemnych. Niedogodności tej można uniknąć przez używanie szesnastkowego formatu zapisu liczb całkowitych.

**ENTIER (X : REAL) : INTEGER**

Największa liczba całkowita mniejsza bądź równa X. Funkcja ta uzupełnia znane w Pascalu funkcje ROUND i TRUNC.

**EOLN : BOOLEAN**

Funkcja EOLN wskazuje na obecność końca linii (bufora wejściowego). Funkcja ma wartość TRUE, gdy następnym znakiem bufora przeznaczonym do czytania jest CHR (13).

**FRAC (X : REAL) : REAL**

Część ułamkowa liczby, tzn. X — ENTIER (X).

**HALT**

Zatrzymanie wykonania programu i wyświetlenie komunikatu

Halt at PC = ....

Procedura ta w połączeniu z wydrukiem kompilacji może służyć do wyszukiwania błędów w działającym programie.

**INCH : CHAR**

Funkcja umożliwiająca czytanie znaków bezpośrednio z klawiatury. Jej wartością jest znak odpowiadający aktualnie wciśniętemu klawiszowi. Jeżeli żaden klawisz nie jest wciśnięty, funkcja przyjmuje wartość CHR (0).

**INLINE (B1. B2, B3...)**

Procedura ta umożliwia wstawianie w tekst programu źródłowego fragmentów zapisanych w kodzie maszynowym; nie jest więc procedurą w pełnym tego słowa znaczeniu, ponieważ działa w czasie kompilacji. Mianowicie ciąg parametrów INLINE zostaje włączony do generowanego w czasie kompilacji kodu w miejscu, w którym INLINE występuje w tekście źródłowym. Ilość parametrów jest dowolna. Powinny one mieć postać liczby z zakresu 0...255 w postaci dziesiętnej lub szesnastkowej.

**INP (PORT : INTEGER) : CHAR**

Funkcja służąca do odczytywania stanu portów procesora bez użycia procedury INLINE. Wartość parametru PORT jest przesyłana do rejestru BC. Wynik funkcji otrzymywany jest następnie poprzez wykonanie instrukcji IN A, (C).

**MARK (VAR VSK : TypWskaźnikowy)**

Zapamiętanie bieżącej zawartości wskaźnika stosu zmiennych dynamicznych przy pomocy zmiennej WSK. Zmienna ta może należeć do dowolnego typu wskaźnikowego. Tego typu markowanie służyć może późniejszemu zwalnianiu pamięci stosu przy pomocy procedury RELEASE.

**OUT (PORT : INTEGER; WART : CHAR)**

Umożliwia korzystanie z portów procesora bez posługiwania się procedurą INLINE. Wynikiem wywołania procedury jest załadowanie wartości PORT do rejestru BC procesora oraz WART do rejestru A. Następnie wykonywana jest instrukcja OUT (C), A.

**PAGE**

Odpowiada wywołaniu WRITE (CHR (12)), tzn. kasuje zawartość ekranu lub wysuwa papier drukarki do nowej strony.

**PEEK (ADR : INTEGER; TYP) : TYP**

Funkcja PEEK służy do odczytywania zawartości określonych bajtów pamięci. Pierwszym jej parametrem jest adres początku badanego obszaru. Drugim, dość nietypowym — specyfikacja typu, w jakim wystąpić ma wynik funkcji. Może to być dowolny typ; dozwolone są np. wywołania

PEEK (32000, ARRAY [1 ...10] OF CHAR)
PEEK #50FF, REAL)
PEEK (ADDR (V), INTEGER)

**POKE (ADR : INTEGER; WART : DowolnyTyp)**

Umieszczenie w pamięci pod adresem ADR wartości wyrażenia WART. WART może należeć do dowolnego typu za wyjątkiem zbiorów. Należy zauważyć, że w większości przypadków umieszczane w pamięci wyrażenie zajmuje więcej niż jeden bajt, np. INTEGER — 2, REAL — 4.

**RANDOM : INTEGER**

Bezparametrowa funkcja losowa, której wynikiem jest liczba całkowita z przedziału 0...255. Funkcja RANDOM jest bardzo szybka, ale przez to również nie najlepiej nadaje się do pewnych zastosowań. Jej wyniki trudno uznać za losowe w przypadku, gdy kolejne wywołania wykonywane są wewnątrz pętli nie zawierającej operacji wejścia/wyjścia.

W tego typu przypadkach należy własnoręcznie napisać funkcję pseudolosową (w Pascalu lub kodzie maszynowym) bardziej dostosowaną do wymagań tworzonych programów.

**READ(..)**

Procedura READ odpowiada standardowi Pascala. Przy pomocy tej procedury można wczytywać z wejścia dane typu CHAR, INTEGER, REAL, ARRAY OF CHAR. Należy jednak pamiętać, że bufor wejściowy na początku wykonania każdego programu zawiera znak CHR (13). Zatem przed wykonaniem pierwszej procedury READ zachodzi warunek EOLN = TRUE. Nie ma to szczególnego znaczenia, gdy pierwszą wczytywaną daną jest liczba. Jeżeli jednak rozpoczęlibyśmy od wczytania zmiennej typu CHAR, to zostanie jej przypisana wartość CHAR (13), zaś następnie pobrana zostanie z klawiatury i umieszczona w buforze nowa linia tekstu. Następna instrukcja READ operuje już na tej nowej linii.

W celu uniknięcia tej mylącej sytuacji należy bezpośrednio przed użyciem pierwszej instrukcji READ użyć READLN. Procedura READLN powoduje zawsze odnowienie zawartości bufora wejściowego przez pobranie nowej linii z klawiatury. Należy też pamiętać o sprawdzaniu warunku EOLN przed wczytywaniem kolejnych znaków, o ile nie wczytujemy ich przy pomocy READLN.

**RELEASE (VAR WSK : TypWskaźnikowy)**

Zwolnienie pamięci stosu zmiennych dynamicznych przez obniżenie jego wierzchołka do wartości wskazywanej przez zmienną wskaźnikową WSK, której wartość została uprzednio nadana przez MARK.

**SIZE (VAR Z : DowolnyTyp): INTEGER**

Parametrem funkcji jest zmienna dowolnego typu, jej wynikiem — liczba całkowita określająca ilość bajtów pamięci zajętych przez tę zmienną.

**TIN (NAZWA : STR; START : INTEGER)**

Procedura wczytuje z taśmy plik o nazwie NAZWA (STR jest typem ARRAY [1 ...8] OF CHAR) i umieszcza jego zawartość w pamięci począwszy od adresu START. Procedura ta w połączeniu z TOUT może służyć do odczytywania z taśmy nagranych uprzednio wartości zmiennych.

**TOUT (NAZWA : STR; START, DLUG : INTEGER)**

Procedura umożliwiająca nagranie na taśmę zawartości określonego obszaru pamięci. Pierwszym parametrem jest nazwa pliku. Typ STR odpowiada tutaj tablicy ARRAY [1...8] OF CHAR. Wywołanie powoduje nagranie na taśmę DLUG bajtów pamięci, począwszy od adresu START. Na przykład zawartość zmiennej X można nagrać przy pomocy TOUT w następujący sposób:

TOUT (,ZM_ X', ADDR (X),SIZE (X));

USER (ADR: INTEGER)

Wywołanie procedury w kodzie maszynowym, znajdującej się w pamięci pod adresem ADR. Procedura ta musi zawierać rozkaz RET procesora (#C9), co jest sprawdzane przez kompilator. Nie może ona również naruszać zawartości rejestru IX.

**WRITE (...)**

Przy pomocy procedury WRITE można drukować dane liczbowe (typów REAL i INTEGER), typu CHAR i ARRAY OF CHAR. Dla wszystkich typów wypisywanych danych wolno określać szerokość pola wydruku, podobnie jak w Pascalu standardowym. Pole to nie może być w przypadku tablic znakowych węższe od długości tablicy.

Wprowadzono dodatkową możliwość wypisywania liczb całkowitych w zapisie szesnastkowym. Czyni się to, podając format:

<liczba>: <szerokość> : H

Na przykład wywołanie

WRITE (1024 : 4 : H); daje w rezultacie wydruk

0400

Oprócz powyższej zmiany, Hisoft Pascal niezależnie od typu komputera, na którym jest zaimplementowany, używa własnego zestawu znaków sterujących wydrukiem Są to:

- CHR (8) — kasowanie znaku poprzedzającego kursor i cofnięcie kursora o jedno miejsce,
- CHR (12) — kasowanie całego ekranu lub nowa strona na drukarce,
- CHR (13) — przejście kursora na początek nowego wiersza,
- CHR (16) — przełączenie wydruku z ekranu monitora na drukarkę i odwrotnie,

W niektórych sytuacjach użyteczne może być wypisywanie znaków poprzez odpowiednią procedurę systemową komputera, która interpretuje znaki sterujące w inny od podanego sposób. W przypadku Spectrum umożliwia to krótka procedura SPOUT, dołączona do pakietu TURTLE — o czym za miesiąc.

*Marek Wyrwidąb*


Commodore 116

# COMMODORE 16 i 116

## czyli pułapki na oszczędnych ciąg dalszy

**Commodore 16 został wprowadzony do produkcji w miejsce przestarzałej już wersji (wycofanej notabene dawno z produkcji) VIC-20. Zgodnie z przewidywaniami firmy komputer ten miał zrobić podobna furorę jak C-64,. jednakże (jak sie później okazało) omal nie doprowadził jej wespół z Commodore PLUS/4 do bankructwa.**

Przyjrzyjmy się bliżej konstrukcji a także zaletom i wadom tych modeli (C-116 stanowi skrzyżowanie wnętrza C-16 i obudowy C-PLUS/4 i jest tańszy ze względu na inny rodzaj zastosowanej w nim klawiatury, oferuje jednakże takie same możliwości jak C-16).

Obudowa oryginalnego C-16 jest dokładnie taka sama jak zastosowana w C-64, różnią się one jedynie kolorem. Pierwsze różnice jakie rzucają się w oczy i są dość niewygodne to rozmieszczenie znaków i liter niezgodnie ze standardem przyjętym dla C-64; tych wszystkich, którzy przesiadają się więc bądź pracowali z C-64 czekają „radosne" pomyłki w początkowym okresie pracy na C-16.

Commodore 16/116 pracują w oparciu o mikroprocesor 7501 stanowiący ulepszoną wersję procesora 6502. Ponieważ lista publikowanych rozkazów tego procesora jest identyczna z poprzednikami nie będą mieli kłopotu ci wszyscy, którzy asembler 6502 poznali na innych maszynach. Oznacza to także, że istnieje wymienność programów maszynowych z innymi modelami Commodore pod warunkiem, że nie będą one się odnosiły do procedur zawartych w ROM.

C-16/116 jest wyposażony w 32 KB ROM oraz 16 KB pamięci RAM, z możliwością rozszerzenia tej ostatniej do 64 KB dwoma sposobami. Pierwszy z nich to zakup odpowiedniego modułu, drugi to względnie mało skomplikowana przeróbka wewnątrz komputera (dla zaawansowanych!) polegająca na wymianie pamięci RAM na układy o większej pojemności oraz przeprowadzeniu (za pomocą kabelków) dodatkowych ścieżek. Odpowiedni schemat był już także opublikowany w „BAJTKU". W pamięci ROM zapisany jest system operacyjny komputera (kernal) oraz BASIC. Po włączeniu komputera do sieci użytkownik ma do swojej dyspozycji ok. 12 KB co w porównaniu z VIC-20 (3.5 KB) jest niewątpliwym krokiem naprzód. Niestety gdy chcemy tworzyć grafikę wysokiej rozdzielczości (HIRES) system operacyjny zabiera nam ok. 10 KB z wolnej pamięci, tak więc użytkownikowi pozostaje jedynie ok. 2 KB co na pewno dużą ilością nie jest i w zasadzie ogranicza wykorzystanie niezłej wersji BASIC do tworzenia dłuższych programów.

### BASIC

C-16/116 pracują w oparciu o wersie BASIC V3.5. która (sprawdzona na znacznie starszych komputerach firmy Commodore np. PET) daje użytkownikowi kilkanaście nowych rozkazów i instrukcji. I tak możliwe jest korzystanie z wpisanego na stałe w ROM monitora języka maszynowego

(MONITOR), instrukcje DRAW, LOCATE, BOX, CIRCLE. PAINT umożliwiają nam znacznie łatwiejsze aniżeli w C-64 operowanie grafiką wysokiej rozdzielczości, dostępne są instrukcje odnoszące się do współpracy ze stacją dysków (DIRECTORY, DLOAD, DSAVE, DVERIFY). Edytor został rozszerzony o kilka bardzo przydatnych rozkazów takich jak AUTO umożliwiającego automatyczne numerowanie wprowadzanych linii programu czy RENUM (ich przenumerowywanie). Zaczątki programowania strukturalnego umożliwiają instrukcje DO... WHILE i DO...UNTIL Można powiedzieć, że w porównaniu z C-64 wersja BASIC jest znacznie lepsza i bardziej bogata co na pewno ułatwia programowanie.

### GRAFIKA

Tak jak w poprzednich modelach, C-16/116 mają do dyspozycji szereg znaczków dzięki którym możemy wykreślić proste rysunki, ramki itp. Komputery te oferują 121 możliwych do wyświetlenia kolorów (15 podstawowych z 8 poziomami luminancji każdy). Grafika wysokiej rozdzielczości podzielona jest na 5 zasadniczych trybów pracy: podstawowego (25 linii na ekranie po 40 znaków — jest to tzw. tryb tekstowy), standardowy tryb graficzny (rozdzielczość 320-200 punktów), standardowy tryb graficzny 320-200 punktów w którym pozostawiono 4 górne linie ekranu do zapisu tekstu, oraz dwa tryby pracy wielokolorowej (multicolor). W tych dwóch ostatnich rozdzielczość obrazu wynosi 160 * 200 punktów i możliwe jest zdefiniowanie do 4 kolorów ze 121 możliwych w każdej matrycy 8*8 punktów. Wszystkie opisane tu tryby pracy (z wyjątkiem tekstowego) zabierają użytkownikowi wspominane wcześniej 10 KB pamięci. Dostępna jest także instrukcja CHAR, umożliwiająca mieszanie tekstu i grafiki na ekranie graficznym.

Jedną z najgorszych wad tych komputerów jest zamiana układu graficznego VIC na TED co ma bardzo duże konsekwencje zarówno w grafice jak i uzyskiwaniu dźwięku. Po pierwsze niemożliwe jest tworzenie sprite-'ów, tak więc oprawa graficzna wielu gier czy programów została żałośnie i bardzo zubożona. Co prawda za pomocą nowego klawisza ESCape możliwe jest tworzenie tzw. „okienek", jednakże tylko jednego w danym czasie; brak jest też odpowiedniej instrukcji w BASIC. Pojawiły się co prawda na rynku nieliczne programy usiłujące złagodzić ten brak, jednakże szkoda ta jest w zasadzie niepowetowana.

### DŹWIĘK

W omawianych komputerach został także zubożony dźwięk — układ 6581, wykorzystywany w wielu profesjonalnych urządzeniach został zastąpiony przez jeden wspólny dla grafiki i dźwięku — TED. Możliwe jest korzystanie z dwóch generatorów dźwięku (w tym jeden z nich może wytwarzać szum) o wspólnie regulowanej głośności. Odpowiednie instrukcje BASIC umożliwiają względnie łatwą obsługę dźwięku. Tego rodzaju „zastępstwo" traktowałbym raczej jako wadę, gdyż znowu pozbawiono użytkownika wielu bardzo ciekawych efektów dźwiękowych do gier czy programów użytkowych. Jak informują podręczniki, wiele możliwości dźwiękowych dostępnych jest z poziomu języka maszynowego, jednakże ogranicza to znacznie liczbę ewentualnych posiadaczy umiejących je wykorzystać we własnym zakresie.

### KLAWIATURA

Pomijając już wspominane powyżej zmiany w umieszczeniu poszczególnych znaków, C-16/116 posiada kilka nowych, bardzo przydatnych funkcji. Po pierwsze użytkownik może zaprogramować sobie dowolnie klawisze funkcyjne za pomocą KEY. Po drugie przybył także nowy klawisz ESCape tworzący wiele nowych funkcji edytora ekranowego (m.in. wspominane już „okienka"). Klawiszom funkcyjnym przypisywane są przez system odpowiednie instrukcje i rozkazy w chwili uruchomienia Komputera.


Commodore 16

## URZĄDZENIA ZEWNĘTRZNE

Pod tym względem C-16/116 nie różni się w zasadzie niczym (z wyjątkiem nieszczęsnych gniazd do joysticków i magnetofonu) od innych komputerów Commodore. Można przyłączać doń stację dysków, drukarkę (port szeregowy — SERIAL PORT), monitor, wzmacniacz m.cz. (gniazdo AUDIOA/VIDEO), dodatkowe moduły takie jak rozszerzenia pamięci (port rozszerzania pamięci — MEMORY EXPANSION). Jest także wyjście antenowe (RF) oraz dwa porty do przyłączania drążków sterowych, a także port magnetofonu.

W tym miejscu znowu należy wypomnieć firmie Commodore wprowadzenie zmian. Zarówno gniazda jak i wtyczki do magnetofonu i joysticków zostały zastąpione innym standardem przez co użytkownik kupując komputer musi jednocześnie postarać się także o zakup odpowiedniego magnetofonu i drążków (w najlepszym wypadku należy kupić odpowiednio złącza pośrednie o ile są dostępne). Jest to czwarta z sześciu dużych wad tego komputera. Złącze magnetofonowe opisywałem już w „BAJTKU" 5/87, niestety znacznie gorzej wygląda sprawa z joystickami (brak schematów, literatury). A może któryś z Czytelników mógłby na tydzień wypożyczyć (lub samemu opisać) takie złącze?

Do rodziny Komputerów C-16/116 i PLUS/4 zapowiadana była także nowa stacja dysków (podobno 3-4 razy szybsza od 1541) o numerze 1551 oraz (ukazała się w nielicznych egzemplarzach) SFD 481. Jak podała firma — stacje te jednak nie nadają się do współpracy z C-64/128.

## LITERATURA I OPROGRAMOWANIE...

czyli piąta i szósta wada tego komputera. Instrukcja firmowa robiła na mnie wrażenie zupełnie nieprzemyślanej i dość niedokładnej; książek o tych modelach ukazało się dość mało i nadal stanowią one rarytas na polskim rynku. Brak dobrej mapy pamięci uniemożliwia wykorzystanie walorów tych komputerów do końca. Polskie pseudotłumaczenia nadają się jedynie do wyrzucenia gdyż nieustannie opisują tylko BASIC i kilka sławetnych POKE np. blokujących wyświetlanie („listowanie") programu. Jak do tej pory nielepiej wygląda sprawa z oprogramowaniem. Potentaci w tej dziedzinie (giełdowi oczywiście) dysponują ok. 300-500 programami dla tych modeli (sprzedawanymi notabene dość słono). Firm piszących oprogramowanie jest coraz mniej zwłaszcza od chwili ukazania się na rynku C-128 i AMIGI.

## *PODSUMOWANIE*

wypada dosyć słabo i moim zdaniem na niekorzyść tego komputera. Ponieważ po artykule o Commodore PLUS/4 kilkunastu Czytelników posiadających już ten sprzęt zarzuciło mi niekompetencję i uprzedzenie do tej rodziny, ocenę końcową tych komputerów pozostawiam już Czytelnikom.

*Klaudiusz Dybowski*

**Zalety:**
- ulepszony edytor ekranowy (w porównaniu do C-64/VIC-20),
- lepsza wersja języka BASIC,
- łatwiejsze programowanie (w BASIC) grafiki wysokiej rozdzielczości,
- niska cena kompletu (z magnetofonem),
- możliwość prostego rozszerzenia pamięci RAM,
- kompatybilność z komputerami C-116 i PLUS/4.

**Wady:**
- niekompatybilność z poprzednimi modelami (zmiana układów, organizacji pamięci),
- zmiana standardu gniazd i wtyków do magnetofonu i joysticków,
- brak sprite'ów,
- duża zajętość pamięci przy korzystaniu z grafiki HIRES,
- duże ograniczenie możliwości dźwiękowych spowodowane wymianą układu SID,
- słaba instrukcja obsługi,
- brak dostępnej powszechnie literatury, zwłaszcza dokładnej mapy pamięci,
- brak powszechnie dostępnego oprogramowania,
- mała ilość firm produkujących software.

# CHAR, RDOT

## CZYLI TEKST I GRAFIKA

**Żalił mi sie ostatnio pewien mój znajomy, że ma kłopoty z wyświetleniem na ekranie wysokiej rozdzielczości wyników obliczeń wraz z wykresem funkcji. Oprócz tego kilku Czytelników miało również wątpliwości jak należy stosować instrukcje CHAR i funkcję RDOT w wyniku czego powstał poniższy program.**

Ponieważ ma on charakter demonstracyjny i składa się z dwóch części proponowałbym przed uruchomieniem postarać się o szklankę herbaty; druga bowiem część tego programu wykonuje się przez ok. 28 minut, ale za to otrzymujemy przepiękny trójwymiarowy wykres funkcji. Program można uruchomić na Commodore 128, 16, 116 i PLUS/4.

Jak wiadomo, instrukcja CHAR umożliwia nam wyświetlenie dowolnego tekstu na ekranie zarówno graficznym jak i tekstowym; parametry ją opisujące są opisane poniżej: CHAR CT, NK, NW, ZT, N

gdzie CT oznacza kolor (0-3), NK — numer kolumny w zakresie 0-79, NW — numer wiersza (0-24), ZT — zmienną tekstową (string), a N obraz odwrócony (negatyw). Przykładowo

CHAR 1, 2, 12, „INSTRUKCJA CHAR", 0 spowoduje wyświetlenie na ekranie napisu INSTRUKCJA CHAR w 12 wierszu (linii) ekranu poczynając od drugiego znaku tego wiersza. Napis ten nie będzie w negatywie (rewersie) gdyż jego znacznik (N) jest ustawiony na 0. Zmiana tego znacznika na 1 spowoduje wyświetlenie napisu w odwróconych kolorach.

Często gdy pracujemy na ekranie graficznym, zależy nam również na wyświetleniu oprócz opisów, także i wyników obliczeń matematycznych. W tym wypadku wyniki te przed wyświetleniem za pomocą CHAR muszą być zamienione z wartości numerycznych na łańcuch (tekst) za pomocą funkcji STR$. Przykładowy program może wyglądać następująco:

```
100 GRAPHIC 1
110 FOR X = 1 TO 180
110 K = SIN/X/*50
120 F$ =STR$/K/
130 CHAR 1, 10, 12, F$, 0
140 NEXT X
```

Bardzo istotny i wygodny dla użytkowników jest fakt, że CHAR można stosować na dowolnym ekranie, także i tekstowym; dzięki temu możemy „zapomnieć" o istnieniu PRINT, TAB i SPC. Tekst wyświetlany za pomocą CHAR jest zawsze zlokalizowany w tym samym miejscu ekranu dopóki nie zmienimy (np. za pomocą pętli) współrzędnych NK i NW:

```
10 TI$ = „000000"
20 CHAR 0, 15, 12, TI$
30 GOTO 20
```

Dodatkowym atutem jest tu fakt, że zegar podany w powyższej postaci można przenieść na ekran graficzny (HIRES) bez żadnych problemów...

W podanym poniżej programie warto zwrócić uwagę na dwie rzeczy. Linie 450 i 460 służą do „formatowania" wydruku — jest to konieczne gdyż aktualnie adresowane punkty ekranu są określane raz dwoma a raz trzema cyframi. Jeżeli np. ostatnią współrzędną trzycyfrową była wartość 101 i następna będzie 95, to w rezultacie, bez formatowania,

na ekranie ukazałaby się liczba 951 czyli aktualna współrzędna i pozostała jeszcze z poprzedniego wyświetlenia jedynka. Aby tego uniknąć, gdy nasza liczba jest dwucyfrowa, program automatycznie dodaje do niej jeden odstęp co powoduje jej przesunięcie na ekranie i skasowanie starej liczby całkowicie.

Linie 320 — 340 (i 540) zawierają procedurę „migania" na ekranie napisu „WCIŚNIJ DOWOLNY KLAWISZ" — można ją z powodzeniem zastosować w innych programach. Zastępując ostatnią jedynkę zerem spowodujemy także migotanie podkładu napisu.

Na zakończenie kilka słów o RDOT. Funkcja ta pozwala nam określić pozycję i kolor źródłowy aktualnie adresowanego na ekranie HIRES punktu graficznego. RDOT/0/ odnosi się zawsze do osi X i podaje wartości w zakresie 0 — 319; RDOT/1/ pozwala nam natomiast odczytać współrzędną Y (zakres 0 — 200). Odczytanie koloru odbywa się za pomocą RDOT/2/.

*Klaudiusz Dybowski*

```
100 REN *** CHAR/RDOT DEMO ***
110 :
120 REM *** C-128/+4/16/116 ***
121 :
130 GRAPHIC 1
140 REM:COLOR 0,1,COLOR 1,15:COLOR 4,8
170 SCNCLR
180 CIRCLE 1,160,100,65
190 CHAR 1,01,24,"WSP. X ="+STR$(160)
200 CHAR 1,14,24,"WSP. Y ="+STR$(100)
210 CHAR 1,27,24,"PROMIEN ="+STR$(65)
220 CHAR 1,01,01, "PRZYKLAD WYKORZYSTANIA
INSTRUKCJI CHAR",1
230 DRAW 1,160,100 TO 225,100
240 DRAW 1,000,189 TO 319,139
250 CHAR 1 , 22, 11 , "R=65" , 0
260 CHAR 1,02,03,"OBWOD KOLA =
2"+CHR$(158)+"R =",0
270 Z$= "2*" +STR$(π)+"*" + STR$(65)
280 CHAR 1,21,03,Z$,1
290 CHAR 1,10,22,"WYNIK  ="+STR$(2*π*65), 1
300 SLEEP 5
310 REM:COLOR 1,8
320 CHAR 1,08,20,"WCISNIJ DOWOLNY
KLAWISZ",1
330 GOSUB 540
340 IF PEEK(208)<>0 GOTO 370
350 CHAR 1,08,20,"
",1: REM: 23 SPACJE
360 GOSUB 540:GOTO 320
370 SCNCLR:TI$="000000"
380 AA=COS(45/180*π):AB=SIN(45/180*π)
390 FOR Z=180 TO -180 STEP-10
400 FOR X=-180 TO 180 STEP2
410 Y=SIN(X/180*π)*SIN(Z/180*π)*90
420 BX=160+.5*(X+Z*AA)
430 BY= 100-.5*(Y+Z*AB)
440 PX=RDOT(0):PY=RDOT(1)
445 PX$=STR$(PX):PY$=STR$(PY)
450 IFLEN(PX$)><4THENPX$=" "+PX$
460 IFLEN(PY$)><4THENPY$=" "+PY$
470 CHAR 1,02,01,"WSP. X ="+PX$
480 CHAR 1,02,02,"WSP. Y ="+PY$
490 CHAR 1,24,01,"CZAS = "+TI$,1
500 DRAW 1,BX,BY+1
510 NEXT X,Z
520 GETKEYA$
530 GRAPHIC 0
535 END
536 :
540 FOR X=1 TO 200:NEXT:RETURN
```

# INSTRUKCJE SSHAPE I GSHAPE DLA C-128/16/116/+4

**Otrzymałem ostatnio od Czytelników sporo listów z prośbą o wyjaśnienie działania instrukcji SSHAPE i GSHAPE. Mam nadzieję, że poniższy artykuł zaspokoi wymagania wszystkich Czytelników.**

Obie te instrukcje są instrukcjami graficznymi i odnoszą się do grafiki wysokiej rozdzielczości (HIRES). SSHAPE pozwala nam Przypisać zmiennej tekstowej dowolnie wybrany obszar ekranu graficznego natomiast GSHAPE umożliwia jego odtworzenie w określonym miejscu. Obszary o których mowa mają zawsze kształt prostokąta lub kwadratu

Dla użytkowników takie rozwiązanie jest bardzo wygodne. Przypuśćmy, że chcemy powielić na ekranie w paru miejscach np. obraz jakiejś figury. W wersjach BASIC pozbawionych tych instrukcji trzeba by było w tym celu stworzyć odpowiedni podprogram figurę tą tworzący, a następnie zmieniając odpowiednio parametry kolejno go wywoływać co jest oczywiście powiązane z wolniejszym wykonywaniem programu i większą zajętością pamięci. Znacznie prostszym sposobem jest tu właśnie zastosowanie SSHAPE oraz GSHAPE. Instrukcje te wymagają określenia następujących parametrów: SSHAPE nz. X1. Y1. (X2,Y2) gdzie:

nz — nazwa zmiennej tekstowej

X1 i Y1 — współrzędne lewego górnego rogu obszaru który chcemy zapisać X2 i Y2 — współrzędne prawego dolnego Rogu tego obszaru Współrzędne XI i Y1 oraz X2 i Y2 muszą zawierać się w przedziale 0,0—320.200. Parametry X2 i Y2 nie muszą być określane (są opcjonalne).

Przykład:

SSHAPE A$,10,10,20,20 — przypisze zmiennej A$ obszar prostokąta określonego współrzędnymi 10,10 (lewy górny róg) i 20,20 (prawy dolny róg). . SSHAPE C$,10,10 — przypisze zmiennej CS obszar o współrzędnych lewego górnego rogu 34.47 do aktualnej pozycji kursora graficznego (czyli do ostatnio adresowanego punktu ekranu HIRES). Pozycję tą możemy odczytać za pomocą funkcji RDOT(0) (współrzędna X) oraz RDOT(1) (współrzędna Y). Niestety nasza róża nie jest pozbawiona kolców. Przy korzystaniu ze SSHAPE należy pamiętać, że zmiennej tekstowej można przypisać normalnie 255 znaków co oznacza, że jej pojemność jest ograniczona. Dozwoloną maksymalnie długość (przy jej przekroczeniu otrzymamy komunikat o błędzie STRING TOO LONG ERROR) możemy obliczyć z następującego wzoru: D = INT ((ABS)X1 - X2) = 1((8 = 0.99) * (ABS) (Y1 - Y2) + 1) + 4

Powyższy wzór odnosi się wyłącznie do standardowego ekranu graficznego: dla trybu wielokolorowego (multicolor) należy wartość 8 zastąpić wartością 4.

# DOS + DLA C–64

**Nagroda za cierpliwe (i bezbłędne!) wpisanie poniższego programu jest uzyskanie trzech dodatkowych rozkazów bardzo przydatnych wszystkim posiadaczom stacji dysków.**

Pierwszy z nich umożliwia wczytywanie katalogu (directory) dyskietki bez jednoczesnego kasowania zawartości pamięci RAM:

$ — umożliwia wczytanie całego katalogu, lub

$ „nazwa 1", „nazwa 2",

co z kolei pozwala nam na wczytanie jedynie określonej grupy tytułów Np.

* „TER*", „??FOX

spowoduje wczytanie wszystkich tytułów których trzy pierwsze znaki stanowią litery TER oraz te tytuły w których znaki trzeci czwarty i piąty są literami FOX Jeżeli użyjemy $ i cudzysłowu /$ „/, to wyświetlona zostanie nazwa dyskietki i ilość wolnych bloków. Wciśnięcie klawisza spacji (odstępu) spowoduje pauzę we wczytywaniu katalogu (wznowienie za pomocą RETURN); Klawisz STOP przerywa wczytywanie na stałe.

Drugi z omawianych rozkazów pozwala na bezpośrednie wczytywanie programów z katalogu dyskietki wyświetlonym na ekranie monitora. Kursor należy umieścić w linii gdzie znajduje się nazwa programu, następnie wciskamy klawisz <— i RETURN. Po wczytaniu podawane są dodatkowo adresy początku i końca obszaru pamięci do jakiego program został wczytany. Rozkaz ten może być używany zarówno w trybie ekranowym (bezpośrednim) jak też i w programie

20 <— „nazwa 1"

Należy jednakże pamiętać, że pierwsza instrukcją tak wczytywanego programu nie może być SYS.

Trzecim rozkazem jest @ . który umożliwia nam szybkie przesyłanie poleceń do stacji dysków (RENAME, SCRATCH, VALIDATE, INITIALIZE, NEW). Po wykonaniu danego polecenia kanał rozkazowy (15) jest automatycznie zamykany i pojawia się komunikat informujący użytkownika o statusie dysku. Status można także uzyskać za pomocą @ lub @ „.

Po wpisaniu programu, należy go najpierw zapisać na dysku/taśmie i dopiero potem uruchomić. Gdy z jakiegoś powodu wykonamy SYS64738 lub wyzerujemy komputer za pomocą kombinacji STOP i RESTORE, to jego ponowne aktywowanie możemy przeprowadzić za pomocą SYS 52550...

*Klaudiusz Dybowski*

```
10 REM *****************
20 REM *               *
30 REM *     DOS +     *
40 REM *               *
50 REM *   W. ZABDYR   *
60 REM *               *
70 REM *****************
80 :
90 FORI=52550TO53204:READQ:POKEI,Q:D=D+Q:NEXT
100 IFD<>85101THENPRINT"[CLR]NIEDOBRE DANE ! ":STOP
110 PRINTCHR$(147)"[CLR]PROGRAM WCZYTANY."
120 SYS52550:NEW
130 '
140 DATA 162,035,189,158,207,032,210,255,202,016,247
150 DATA 169,109,141,008,003,169,205,141,009,003,096
160 DATA 169,228,141,008,003,169,167,141,009,003,096
170 DATA 032,121,000,076,231,167,162,001,032,115,000
180 DATA 201,064,240,087,201,036,240,127,201,095,208
190 DATA 233,232,142,157,207,162,016,142,208,206,162
200 DATA 004,032,115,000,201,034,240,008,202,208,246
210 DATA 162,011,076,139,227,162,000,032,187,206,224
220 DATA 000,208,005,162,008,076,139,227,032,219,206
230 DATA 169,000,032,213,255,165,144,201,064,240,003
240 DATA 032,003,207,196,056,144,008,240,002,176,008
250 DATA 228,055,176,004,134,045,132,046,032,060,207
260 DATA 076,165,206,169,076,141,056,207,169,111,141
270 DATA 157,207,032,115,000,240,026,201,034,208,179
280 DATA 032,013,207,230,125,162,000,032,187,206,198
290 DATA 125,032,174,255,165,157,208,003,076,168,206
300 DATA 076,024,207,141,213,207,169,038,141,208,206
310 DATA 169,000,141,157,207,032,115,000,240,013,201
320 DATA 034,208,132,169,058,141,214,207,232,032,187
330 DATA 206,032,219,206,169,147,032,210,255,160,007
340 DATA 032,014,229,160,003,032,136,207,162,002,032
350 DATA 095,207,162,028,032,228,255,032,210,255,202
360 DATA 208,247,032,136,207,162,003,032,095,207,133
370 DATA 099,032,228,255,133,098,032,102,207,160,011
380 DATA 032,014,229,032,228,255,166,144,208,066,201
390 DATA 034,208,245,032,210,255,032,228,255,032,210
400 DATA 255,201,034,208,246,160,030,032,014,229,032
410 DATA 228,255,201,032,240,249,032,210,255,032,228
420 DATA 255,208,248,169,013,032,210,255,165,197,201
430 DATA 064,240,014,032,234,255,032,225,255,240,021
440 DATA 165,197,201,001,208,242,165,144,240,163,162
450 DATA 015,189,193,207,032,210,255,202,016,247,032
460 DATA 149,207,162,000,134,198,169,096,141,056,207
470 DATA 165,157,240,003,076,116,164,076,174,167,160
480 DATA 000,132,131,032,115,000,157,213,207,240,015
490 DATA 201,034,240,008,232,032,056,207,224,038,144
500 DATA 236,032,115,000,169,239,133,131,096,138,162
510 DATA 213,160,207,032,189,255,169,001,162,008,172
520 DATA 157,207,032,186,255,032,192,255,162,001,032
530 DATA 198,255,032,228,255,133,099,032,228,255,133
540 DATA 098,165,144,240,215,032,149,207,162,004,104
550 DATA 104,076,139,227,169,008,032,177,255,173,157
560 DATA 207,076,147,255,169,008,032,180,255,173,157
570 DATA 207,032,150,255,169,018,032,210,255,032,165
580 DATA 255,032,210,255,201,013,208,246,032,171,255
590 DATA 076,168,206,096,168,255,096,165,157,240,154
600 DATA 138,072,152,072,169,013,032,210,255,169,018
610 DATA 032,210,255,032,107,207,169,045,032,210,255
620 DATA 104,133,098,104,133,099,076,107,207,032,228
630 DATA 255,202,208,250,096,160,007,032,014,229,162
640 DATA 144,056,032,073,188,032,223,189,032,135,180
650 DATA 032,130,183,152,170,160,000,177,034,032,210
660 DATA 255,200,202,208,247,096,162,001,185,209,207
670 DATA 032,210,255,136,202,016,246,096,032,204,255
680 DATA 169,001,076,195,255,000,013,146,032,055,056
690 DATA 057,049,032,041,067,040,032,032,082,089,068
700 DATA 066,065,090,032,046,087,032,032,032,042,032
710 DATA 043,083,079,068,032,042,032,018,013,046,072
720 DATA 067,089,078,076,079,087,032,087,079,075,079
730 DATA 076,066,013,032,032,018
```


http://www.t2e.pl/reduksy

Instrukcja GSHAPE jest odwrotnością SSHAPE czyli umożliwia nam odtworzenie na ekranie wysokiej rozdzielczości zapamiętanego uprzednio obrazu. GSHAPE wymaga jedynie określenia współrzędnych lewego górnego rogu ekranu gdzie ma być nasz obraz wyświetlany oraz dodatkowo (aczkolwiek niekoniecznie) trybu pracy:

GSHAPE A$,X1,Y1,t
Można wybrać następujące tryby pracy:
0 — obraz zostanie wyświetlony tak jak został apamiętany
1 — obraz zostanie wyświetlony w rewersie (zamiana kolorów tła i rysunku)
2 — obraz poddawany jest operacji logicznej OR z podkładem na którym będzie wyświetlony
3 — jak wyżej z tym, że poddany będzie operacji logicznej AND
4 — jak wyżej z tym, że poddany będzie operacji logicznej EXOR Gdy tryb nie zostanie określony, komputer przyjmuje, ze chodzi tu o tryb 0.

Zalecałbym ostrożne korzystanie z trybów 1—4 gdy pracujemy w trybie wielokolorowym (multicolor) gdyż możemy otrzymać wtedy dość nieoczekiwane efekty a to ze względu na inne funkcje przypisywane poszczególnym bitom składającym się na pojedynczy punkt ekranu. GSHAPE DD$,15,34 — spowoduje wyświetlenie uprzednio zapamiętanego obrazu przypisanego zmiennej DD$ w miejscu o współrzędnych lewego górnego rogu obrazu 15,34.

GSHAPE TWS 100,100,1 — podobnie jak wyżej z tym, że obraz będzie wyświetlony w rewersie.

Na zakończenie przedstawiam mały program demonstracyjny, pokazujący w jaki sposób można korzystać z opisanych w artykule instrukcji. Linia 120 powoduje skasowanie i wyczyszczenie pamięci ekranu graficznego. Następnie (linia 125) w lewym rogu ekranu wykreślane jest małe kółko, które zostaje przypisane zmiennej A$ (linia 130). SCNCLR w linii 140 powoduje ponowne wyczyszczenie ekranu i odtworzenie uprzednio zapamiętanego pod zmienną A$ obrazu kółka. Stosując pętle otrzymujemy w rezultacie na ekranie szereg kółek. Firma Commodore zapewnia, ze BASIC V4.5 komputerów C-16/116 i PLUS/4 jest całkowicie kompatybilny z wersją V7.0 Commodore 128, tak więc użytkownicy tych modeli nie powinni mieć żadnych problemów z uruchomieniem programu choć był on przygotowywany na C-128. Życzę przyjemnej zabawy...

*Klaudiusz Dybowski*

```
100 REM *** SSHAPE/GSHAPE DEMO
***
105 :
110 REM *** COMMODORE 128 ***
111 :
112 :
115 GRAPHIC 1
120 SCNCLR
125 CIRCLE 1,10, 10, 10, 10
130 SSHAPE A$,0,0,2 0,20
135 SLEEP 3
140 SCNCLR
145 FOR X=1 TO 319 STEP 10
150 GSHAPE A$,X,1+X/2
155 NEXT
160 FOR X=1 TO 319 STEP 10
165 GSHAPE A$,X/5,X/2
170 NEXT
175 GSHAPE A$,100,100,1
180 SLEEP 5
185 GRAPHIC 0
190 SCNCLR : PRINT "KONIEC. "
```

# BAJTEK DLA COMMODORE 64

**Trudno mi było wybrać lepsza nazwę dla tego programiku zajmującego 38 bajtów pamięci od adresu 680 do 717. Program ten może być pomocny przy wszelkich pracach związanych z operacjami na bitach takimi jak projektowanie własnej czcionki i kroju liter, znaków dla drukarki itp.**

Zadaniem głównym programu jest określenie, które bity są ustawione (włączone) i jaka jest ich wartość dla danej wartości dziesiętnej w zakresie od 0 do 255. W tym celu należy ową wartość wpisać za pomocą POKE do komórki 2 i wykonać SYS 680. Pod spodem ukaże się „czarno na białym" (czyli w rewersie), które z bitów są ustawione (wartości większe od zera), a które są wyłączone (równe 0). Z programu można korzystać zarówno w trybie ekranowym (bezpośrednim) jak też i w trybie programowym. „BAJTEK 64" zawiera w sobie także krótką demonstrację, nie powinno być więc kłopotu z jego wykorzystaniem. Poniżej przedstawiono również procedurę w języku wewnętrznym realizującą opisywaną funkcję. Gdy z jakichś powodów adres 680 ($02A8) nie będzie użytkownikowi odpowiadał, program można przenieść w dowolny obszar pamięci (jest on w pełni relokowalny). „BAJTEK 64" wykorzystuje komórki na stronie zerowej o adresach 2, 251 i 252 ($02, $FB i FC).

Tu na marginesie pewna uwaga. Choć z pozoru wydawać by się mogło, że zapisywanie rejestru Y w pamięci jest zbędne, okazało się to konieczne ze względu na procedurę PRTLIN (JSR 48589, $BDCD), która zmienia zawartość tego rejestru. Posiadacze innych typów Commodore zmuszeni będą niestety zastosować drogę okrężną (wyświetlanie na ekranie poprzez CHROUT/BSOUT, a nie PRTLIN) gdyż w nowszych wersjach Commodore została ona nieco zmieniona; jest to o tyle wygodniejsze, że CHROUT/ BSOUT zabezpiecza wszystkie wartości znajdujące się aktualnie w poszczególnych rejestrach przed jej wykonaniem, tak więc zapis rejestru Y w pamięci nie będzie konieczny.

*Klaudiusz Dybowski*

```
100 REM *** BAJTEK C-64 ***
110 :
120 :
130 FORJ=680TO717:READA:POKEJ,A:C=C+A:NEXT
140 IFC<>5906THENPRINT"QNIEDOBRE DANE !":STOP
150 PRINTCHR$(147)"QPROGRAM WCZYTANY. "
160 PRINT"QQDEMO  :Q":FORI=1TO255STEP25:POKE2,I
165 PRINTI;TAB(5)" =  ";:SYS680:NEXT:NEW
170 :
180 DATA 169,128,133,251,168,200,132,199,132,252,165
190 DATA 002,037,251,170,169,000,032,205,189,169,032
200 DATA 032,210,255,070,251,164,252,192,136,208,228
210 DATA 169,013,076,210,255
```

```
02R8  A9  80     LDR  #$80
02RR  85  FB     STA  $FB
02RC  A8         TAY
02RD  C8         INY
02RE  84  C7     STY  $C7    ;RVS ON
02B0  84  FC     STY  $FC    ;ZAPISZ .Y
02B2  A5  02     LDA  $02    ;WCZYTAJ LICZBE
02B4  25  FB     AND  $FB    ;POROWNAJ BITY
02B6  AA         TAX          WYSWIETL WYNIK
02B7  A9  00     LDA  #$00   ;
02B9  20  CD BD  JSR  $BDCD;
02BC  A9  20     LDA  #$20   ;I ODSTEP
02BE  20  D2 FF  JSR  $FFD2
02C1  46  FB     LSR  $FB    ;PRZESUN W LEWO
02C3  A4  FC     LDY  $FC    ;WCZYTAJ .Y
02C5  C0  88     CPY  #$88   ;8 OBIEG PETLI?
02C7  D0  E4     BNE  $02AD; NIE
82C9  A9  0D     LDA  #$0D   ;LINIA ODSTEPU
02CB  4C  D2 FF  JMP  $FFD2
```

## Co piszczy pod klawiaturą?

**(cz. 11)**

# TABLICA ADRESÓW PROCEDUR SYSTEMOWYCH c.d.

| Nr | Adres wektora | Adres rzeczywisty/opis | | |
|---|---|---|---|---|
| | | 464 | 664 | 6128 |
| 140 | BCA4 | 2851 | 29C1 | 29C1 |
| | | Weryfikacja (porównanie zapisu na kasecie z zawartością pamięci). Wej: HL zawiera adres danych do porównania, DE zawiera ilość bajtów do porównania, A zawiera znak synchronizacji. Wyj: Jeżeli weryfikacja przebiegła prawidłowo to CARRY= 1. W przypadku przeciwnym CARRY = 0 a A zawiera kod błędu. W każdym przypadku AF, BC, DE, HL i IX są modyfikowane. | | |
| 141 | BCA7 | 1E68 | 1FE9 | 1FE9 |
| | | Inicjalizacja obsługi toru akustycznego. Wej: niema Wyj: AF, BC, DE, i HL są modyfikowane. | | |
| 142 | BCAA | 1F9F | 2114 | 2114 |
| | | Dokładanie dźwięku do kolejki (queue) Wej: HL zawiera adres bloku 9 bajtów definiujących wymagany dźwięk. Wyj: Jeżeli do kolejki ma być dodany dźwięk to CARRY=1 i HL są modyfikowane. Jeżeli kolejka jest wypełniona i dźwięk nie może byc dodany to CARRY = 0 i HL są zabezpieczane. W każdym przypadku AF, BC, DE i IX są modyfikowane. Pozostałe rejestry są zabezpieczane. | | |
| 143 | BCAD | 206C | 21CE | 21CE |
| | | Sprawdzenie czy istnieje miejsce w kolejce dźwiękowej. Wej: A zawiera numer testowanego kanału (1 dla A, 2 dla B i 4 dla C). Wyj: A zawiera stan testowanego kanału. F, BC, DE i HL są modyfikowane. | | |
| 144 | BCB0 | 2089 | 21EB | 21EB |
| | | Przygotowanie wykonania przerwania w momencie wolnej kolejki dźwiękowej. Wej: A zawiera numer oczekującego kanału, HL zawiera adres bloku pro gramu przerwania. Wyj: AF, BC, DE i HL są modyfikowane. | | |
| 145 | BCB3 | 204A | 21AC | 21AC |
| | | Zapis punktu na ekranie bez uwzględnienia trybu określonego w poprzedniej (115) procedurze. Wej: B zawiera maskę atramentu, C zawiera maskę punktu, HL zawiera adres punktu w pamięci. Wyj: AF są modyfikowane | | |
| 146 | BCB6 | 1ECB | 2050 | 2050 |
| | | Zatrzymanie dźwięków we wszystkich kanałach. Wej: niema Wyj: Jeżeli jakiś kanał był aktywny to CARRY = 1. W przeciwnym przypadku CARRY = 0. W obu przypadkach AF, BC i HL są modyfikowane. | | |
| 147 | BCB9 | 1EE6 | 206B | 206B |
| | | Uaktywnienie wszystkich dźwięków, które były zatrzymane przez poprzednią (146) procedurę. Wej: nie ma Wyj: AF, BC, DE i IX są modyfikowane. | | |

*Wojciech Ziółek*

# FIRMA • FIRMA •

**Zostałeś dyrektorem nowej firmy. Masz do dyspozycji jedną fabrykę i 50 tys. dolarów kapitału zakładowego. Możesz doprowadzić swoją firmę do rozkwitu, jeśli okażesz się dobrym biznesmenem, albo też stracić pracę, gdy spowodujesz bankructwo przedsiębiorstwa.**

Gry towarzyskie takie jak „Bankrut", „Giełda", „Tranzyt" itp., polegające na modelowaniu mechanizmów wielkiego biznesu, cieszą się od lat powodzeniem nie tylko w naszym kraju. Z jednej strony pozwalają nam wczuć się w rolę rekinów finansjery, dysponujących milionowymi fortunami, a z drugiej, stanowią znakomite pole do wypróbowania umiejętności strategicznego myślenia.

Autor programu „Firma" wykorzystał ten pomysł, uzupełnił o własne przemyślenia, i tak w efekcie powstała znakomita gra komputerowa.

Gra przeznaczona jest w zasadzie dla jednej osoby, która w ciągu 24 miesięcy (oczywiście komputerowe miesiące upływają znacznie szybciej niż w życiu) ma zarobić możliwie najwięcej pieniędzy. Zarabianie pieniędzy wymaga oczywiście inwestycji Trzeba kupować półfabrykaty, które w fabrykach przerabiane są na gotowe wyroby, budować nowe fabryki, płacić za transport. Stosunkowo najmniej zachodu wymaga spekulacja na giełdzie, lecz — z drugiej strony — jest to bardzo niepewny interes.

Można także wziąć kredyt z banku Należy jednak pamiętać, że po sześciu miesiącach, cały kredyt (wraz z odsetkami) trzeba zwrócić. Jeśli kapitał zakładowy jest niższy niż należna płatność, prowadzi to nieuchronnie do bankructwa.

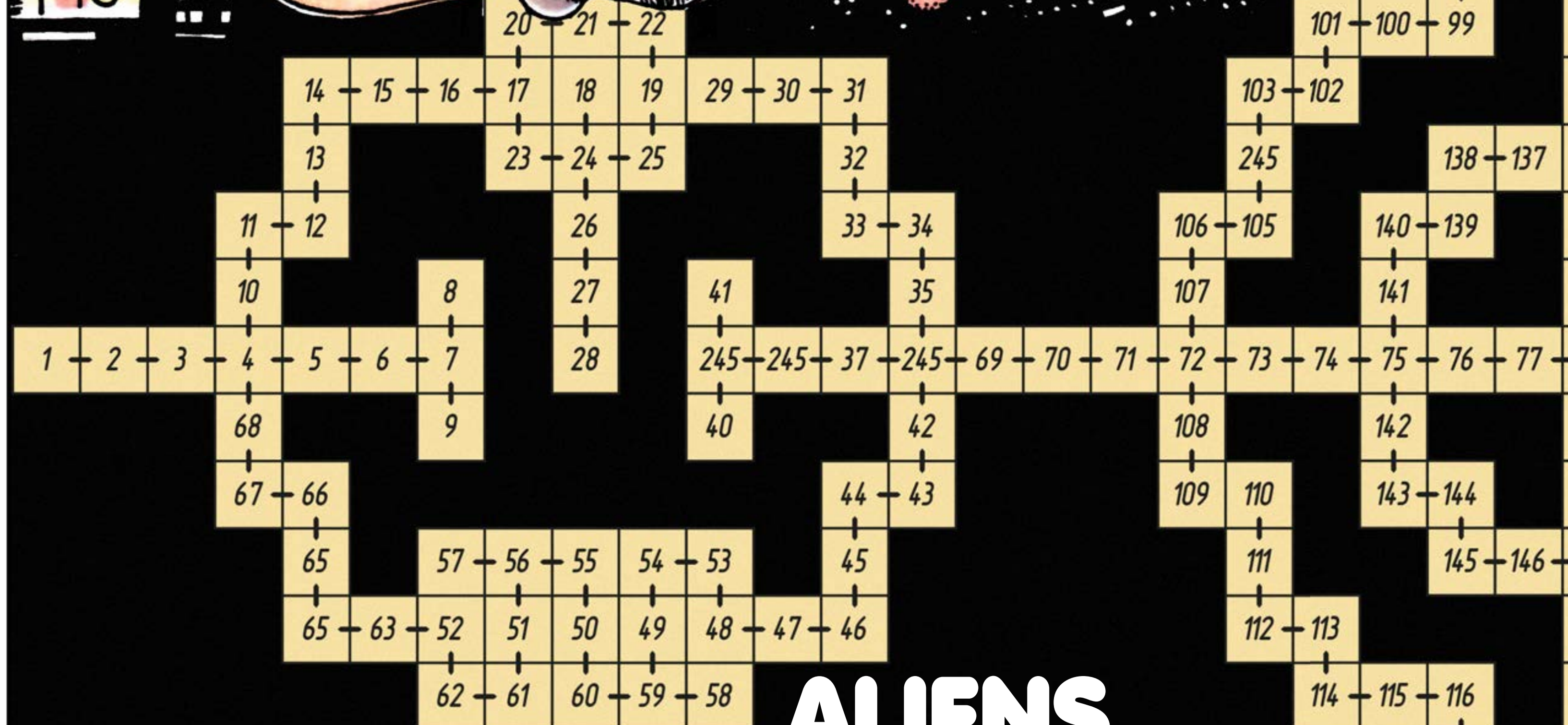
*(rp)*

Autorem programu jest Tomasz Nowicki

```
10 '*********************************
20 '*********************************
30 '*****                       *****
40 '*****        F I R M A      *****
50 '*****                       *****
60 '*********************************
70 '*********************************
80 '
90 GOSUB 3470
100 'T. NOWICKI   1986
110 '
115 ON BREAK CONT
120 ' INSTRUKCJA
130 INK 0,10:INK 1,26:INK 2,6:INK 3,18:MODE
1:PEN 1:PAPER 0
140 BORDER 10: LOCATE 14,1:PRINT"F I R M A"
150 LOCATE 3,3:PRINT"Zosta"+
CHR$(254)+"e"+CHR$(248)+" mianowany dyrektorem
nowo"
160 PRINT"otwartej ";:PEN 2:PRINT"Firmy";:PEN
1:PRINT". Twoim zadaniem jest"
170 PRINT"osi"+ CHR$(240)+"gn"+ CHR$(240) +
CHR$(246)+" maksymalny zysk w okresie 24 "
180 PRINT"miesi"+ CHR$(242)+"cy. ":PRINT "Jako
nowoczesny dyrektor, w swojej pracy pos"+
CHR$(254)+"ugujesz si"+ CHR$(242);:PEN 3:PR
INT" KOMPUTEREM,";:PEN 1:PRINT " kt"+
CHR$(244)+"ry automatycznie wykonuje Twoje
zlecenia."
190 PRINT"Do Ciebie nale"+ CHR$(252)+"y wyb"+
CHR$(244)+"r odpowiedniej":PEN 2:PRINT:PRINT
TAB(7)"S T R A T E G I I  G R Y"
200 PEN 1:PRINT:PRINT"Mo"+ CHR$(252)+"esz
kupowa"+ CHR$(246)+" i sprzedawa"+ CHR$(246)+"
potrzebne do produkcji cz"+ CHR$(242)+ CHR$
(248)+"ci, mo"+ CHR$(252)+"esz spekulowa"+
CHR$(246)+"    akcjami na gie"+
CHR$(254)+"dzie, masz te"+ CHR$(252)+"";
210 PRINT" mo"+ CHR$(252)+"liwo"+ CHR$(248)+
CHR$(246)+" R A Z  w czasie gry zaci"+
CHR$(240)+"gn"+ CHR$(240)+ CHR$(246)+" kredyt,
k
t"+ CHR$(244)+"ry mo"+ CHR$(252)+"esz
dowolnie wykorzy- sta"+ CHR$(246)+". Czas
sp"+CHR$(254)+"aty - 6 mies."
220 LOCATE 12,25:PRINT"Press any key"
230 IF INKEY$="" THEN 230
240 CLS: LOCATE 13,1:PEN 2:PRINT"K O M P U
T E R":PEN 1:PRINT:PRINT:PRINT TAB(6)"Wyb"+
CHR$(244)+"r operacji - w/g MENU"
250 PRINT:PRINT TAB(3)"Komputer automatycznie
koryguje poda- n"+CHR$(240)+" przez Ciebie
wielko"+CHR$(248)+CHR$(246)+" wykonywanej"
260 PRINT" aktualnie operacji, je"+
CHR$(248)+"li za"+ CHR$(248)+" operacja jest
niemo"+ CHR$(252)+"liwa - wraca do MENU"
270 PRINT:PRINT" Liczba 0 - rezygnujesz z
operacji":PRINT:PRINT"Liczba < 0 1 lub "+
CHR$(34)+ CHR$(45)+ CHR$(34)+" - zlecasz
kompute
ro- wi maksymaln"+ CHR$(240)+" warto"+
CHR$(248)+ CHR$(246)+" transakcji"
280 PRINT"/operacja TRANSPORT - zlecasz
transport do wszystkich czynnych fabryk"
290 LOCATE 12,25:PRINT"Press any key"
300 IF INKEY$="" THEN 300
310 CLS: LOCATE 10,1:PRINT"INFORMACJE
DODATKOWE"
320 LOCATE 2,4:PRINT"Ceny zmienne losowo z
tendencj"+ CHR$(240)
330 PRINT" do wzrostu"
340 PRINT:PRINT" Koszt utrzymania jednej
czynnej fabry- ki  - 1000 $ miesi"+
CHR$(242)+"cznie"
350 PRINT:PRINT" Ceny akcji zmienne losowo ze
znacznym   rozrzutem warto"+ CHR$(248)+"ci"
360 PRINT:PRINT" Cena transportu dotyczy
przewozu jed-   nej partii cz"+ CHR$(242)+
CHR$(248)+"ci do jednej fabryki"
370 LOCATE 9,20:PEN 3:PRINT"P O W O D Z E N I
A !":PEN 1
380 LOCATE 12,25:PRINT"Press any key"
390 IF INKEY$="" THEN 380
400 RANDOMIZE TIME
410 '
420 '
430 '
440 MODE 1
450 INK 0,15:INK 1,1:INK 2,18:INK 3,6
460 WINDOW 1,32,20,25
470 WINDOW #1,1,40,1,19
480 WINDOW #2,33,40,20,25
490 BORDER 9:PAPER 3:PEN 2
500 PAPER #1,1:PEN #1,2
510 PAPER #2,0:PEN #2,1
520 CLS:CLS#1:CLS#2
530 READ ccze,ctra,cpro,cinw,kon,fabcz,fab,c
zas, icze,ipro,fbud,ilkr,ltred,czaskr,ceakc,
ilakc
540 DATA 1000,1000,3000,100000,50000,1,1,1,0,0
,0,0,0,0,25000,0
550 DIM a(10,3),b(10)
560 DEF FNt(x)=x-INT(RND*0.15*x)
570 DEF FNp(x,y)=MIN(y,50-a(x-1,1),100-(a(x-
1,1)+a(x-1,2)+a(x-1,3)),icze)
580 DEF FNg(x)=INT(0.9*x+RND*0.25*x)
590 '
600 '
610 '**** POLE GRY ****
620 '
630 LOCATE #1,14,1:PRINT#1,"F I R M A"
640 PEN#1,3:LOCATE #1,4,2:PRINT#1,"MAGAZYN":
LOCATE #1,33,2:PRINT#1,"CENY":PEN#1,2
650 PAPER #1,3:LOCATE #1,14,3:PRINT#1,"KONTO
$":PAPER #1,1: LOCATE #1,29,4:PRINT#1,"Cz";CHR$
(242)+CHR$(248);"ci"
660 LOCATE #1,2,5:PRINT#1,"Cz";CHR$(242)+ CHR$(
248);"ci";TAB(27)"Produkty"
670 LOCATE #1,2,6:PRINT#1,"Produkty";TAB(26)"
Transport"
680 PEN#1,3:LOCATE #1,16,7:PRINT#1,"MIESI";CHR
$(241);"C":PEN#1,2:LOCATE#1,27,7:PRINT#1,"Budo
wa": LOCATE #1,2,7:PRINT#1,"Akcje"
690 '
700 '
710 BORDER 9: PAPER #1,3:LOCATE #1,20,3:PRINT#1
,USING"#######";kon:PAPER #1,1
720 LOCATE #1,11,5:PRINT#1,USING"####";icze
730 LOCATE #1,36,4:PRINT#1,USING"####";ccze
740 LOCATE #1,11,6:PRINT#1,USING"####";ipro
750 LOCATE #1,36,5:PRINT#1,USING"####";cpro:
```

```
LOCATE #1,36,6:PRINT#1,USING"####";ctra
760 PEN#1,3: LOCATE #1,24,7:PRINT
#1,USING"##";czas:PEN#1,2:LOCATE
#1,11,7:PRINT #1,USING"####";ilakc: LOCATE
#1,34,7:PRINT#1,USING"#
#####";cinw
770 FOR i=1 TO fabcz
780 LOCATE #1,5,i+8
790 PRINT#1, USING"FABRYKA ##      ##    ##
##";i,a(i-1,1),a(i-1,2),a(i-1,3):NEXT
800 IF fab>fabcz THEN 810 ELSE 840
810 FOR i=i TO fab
820 LOCATE #1,5,i+8
830 PEN#1,1:PAPER#1,3:PRINT#1,USING"FABRYKA
##   BUDOWA # MIESI"+ CHR$(241)+"C";i,b(i-
1):NEXT:PEN#1,2:PAPER#1,1
840 IF kred>0 THEN LOCATE #1,5,i+8:PAPER
#1,0:PEN#1,1:PRINT#1,USING"KREDYT  #
MIES. SUMA ########";czaskr,kred:PAPER
#1,1:PEN#1,2:EL
SE PRINT#1, TAB(5);SPACE$(30)
850 CLS:BORDER 9:PEN 2:LOCATE 14,1:PRINT "<
MENU >"
860 LOCATE 2,2:PRINT"<";:PEN 2:PRINT"K";:PEN
1:PRINT"UPNO CZ"+ CHR$(243)+ CHR$(249)+"CI>"
870 LOCATE 16,2:PRINT"<";:PEN 2:PRINT"S";:PEN
1:PRINT "PRZEDA"+ CHR$(253)+" CZ"+ CHR$(243)+
CHR$(249)+"CI>"
880 LOCATE 2,3:PRINT"<";:PEN 2:PRINT"T";:PEN
1:PRINT"RANSPORT><SPRZEDA"+ CHR$(253);:PEN
2:PRINT" P";:PEN 1:PRINT"RODUKT"+ CHR$(245)+
"W"
890 LOCATE 2,4:PRINT"<";:PEN 2:PRINT"B";:PEN
1:PRINT"UDOWA><";:PEN 2:PRINT"G";:PEN 1:PRINT
"IE"+CHR$(255)+"DA><KRE";:PEN 2:PRINT"D";
:PEN 1:PRINT"YT><";:PEN 2:PRINT"C";:PEN
1:PRINT"ZAS>"
900 LOCATE 2,5:PRINT"<KA";:PEN 2:PRINT"L";:PEN
1:PRINT"KULATOR> <";:PEN 2:PRINT"Z";:PEN
1:PRINT"AKO"+CHR$(239)+"CZENIE GRY>"
910 PEN 2:LOCATE 10,6:PRINT"TWOJA DECYZJA
?":PEN 1
920 '
930 CLEAR INPUT
940 d$=INKEY$: IF d$="" THEN 940
950 IF d$="k" OR d$="K" THEN 1080
960 IF d$="s" OR d$="S" THEN 1310
970 IF d$="p" OR d$="P" THEN 1440
980 IF d$="t" OR d$="T" THEN 1560
990 IF d$="b" OR d$="B" THEN 1870
1000 IF d$="d" OR d$="D" THEN 2010
1010 IF d$="g" OR d$="G" THEN 2140
1020 IF d$="l" OR d$="L" THEN 2530
1030 IF d$="c" OR d$="C" THEN 2930
1040 IF d$="z" OR d$="Z" THEN 3340
1050 GOTO 930
1060 '
1070 '
1080 ' KUPNO CZESCI
1090 '
1100 '
1110 CLS#2:LOCATE#2,2,1:PRINT#2,"Zakup": LOCA
TE#2,2,3:PRINT#2,"Cz"+CHR$(242)+CHR$(248)+"ci
":BORDER 2
1120 IF ccze>kon THEN PRINT CHR$(7):CLS#2:
GOTO 850
1130 CLS: LOCATE 2,2
1140 PRINT"Ile cz" + CHR$(242)+CHR$(248)+"ci
kupujesz ? ";:GOSUB 1240
1150 IF x<0 OR x>INT(kon/ccze) THEN x=INT(kon/
ccze)
1160 LOCATE #2,3,5:PRINT#2,x
1170 IF x=0 THEN 850
1180 icze=icze+x
1190 kon=kon-x*ccze
1200 GOTO 710
1210 '
1220 'PROCESOR < INPUT >
1230 '
1240 PEN 2:INPUT "",x$:PEN 1:IF x$="" THEN
x$="0"
1250 IF ASC(x$)=45 THEN x=1E+12: RETURN
1260 IF ASC(x$)<48 OR ASC(x$)>57 THEN x$="0"
1270 x=INT(VAL(x$))
1280 RETURN
1290 '
1300 '
1310 'SPRZEDAZ CZESCI
1320 CLS#2: LOCATE #2,1,1:PRINT#2,"Sprzeda"+
CHR$(252): LOCATE #2,2,3:PRINT#2,"Cz"+
CHR$(242) + CHR$(248)+"ci":BORDER 17
1330 IF icze<=0 THEN PRINT CHR$(7):CLS#2:
GOTO 850
1340 CLS:LOCATE 2,2:PRINT"Ile
cz"+CHR$(242)+CHR$(248)+"ci sprzedajesz ? ";:
GOSUB 1240
1350 IF x=0 THEN LOCATE #2,3,5:PRINT#2 ,x:
GOTO 850
1360 IF x>icze THEN x=icze
1370 LOCATE #2,3,5:PRINT#2,x
1380 icze=icze-x
1390 kon=kon+x*ccze
1400 GOTO 710
1410 '
1420 'SPRZEDA PRODUKTOW
1430 '
1440 CLS#2: LOCATE #2,1,1:PRINT#2,"Sprzeda"+
CHR$(252): LOCATE #2,2,2:PRINT#2,"Produk-
':LOCATE #2,2,3:PRINT#2,"t" + CHR$(244)+"w":BO
RDER 19
1450 IF ipro<=0 THEN PRINT CHR$(7):CLS#2 :
GOTO 850
1460 CLS:LOCATE 2,2:PRINT"Ile
produkt"+CHR$(244)+"w sprzedajesz ? ";: GOSUB
1240
1470 IF x=0 THEN LOCATE #2,3,5:PRINT#2, x:
GOTO 850
1480 IF x>ipro THEN x=ipro
1490 LOCATE #2,3,5:PRINT#2,x
1500 ipro=ipro-x
1510 kon=kon+x*cpro
1520 GOTO 710
1530 '
1540 ' TRANSPORT
1550 '
1560 CLS#2: LOCATE #2,2,2:PRINT#2,"Trans":PRIN
T#2,TAB(2)"port":BORDER 16
1570 IF icze=0 OR ctra>kon THEN PRINT
CHR$(7):CLS#2: GOTO 850
1580 CLS: LOCATE 3,2:PRINT"Do kt"+
CHR$(244)+"rej fabryki ? ";: GOSUB 1240
1590 IF x=0 THEN 850
1600 fabnr=x
1610 LOCATE 3,4:PRINT"Ile cz"+ CHR$(242) +
CHR$(248)+"ci ? ";: GOSUB 1240
1620 IF x=0 THEN 850
1630 pal=x
1640 PRINT fabnr:PRINT pal
1650 pal2=FNp(fabnr,pal): IF pal2=0 THEN PRINT
CHR$(7):GOTO 1580
1660 pal3=FNt(pal2)
1670 a(fabnr-1,1)=a(fabnr-1,1)+pal3
1680 kon=kon-ctra:icze=icze-pal2
1690 LOCATE #2,1,5:PRINT#2,"Dotar"+
CHR$(254)+"o":LOCATE #2,1,6:PRINT#2,pal3
1700 IF icze=0 OR ctra>kon THEN PRINT
CHR$(7):CLS#2:GOTO 710
1710 GOTO 710
1720 dosz=0
1730 FOR f=1 TO fabcz
1740 IF kon<ctra OR icze=0 THEN PRINT CHR$(7):
GOTO 1810
1750 pal2=FNp(f,pal):IF pal2=0 THEN PRINT
CHR$(7): GOTO 1810
1760 pal3=FNt(pal2)
1770 a(f-1,1)=a(f-1,1)+pal3
1780 icze=icze-pal2
1790 kon=kon-ctra
1800 dosz=dosz+pal3
1810 NEXT f
1820 LOCATE #2,1,5:PRINT#2,"Dotar"+
CHR$(254)+"o":LOCATE #2,1,6:PRINT#2,dosz
1830 GOTO 710
1840 '
1850 ' BUDOWA
1860 '
1870 BORDER 10:CLS#2:LOCATE #2,2,2:PRINT #2,
"Budowa"
1880 IF kon<cinw OR fab=10 THEN PRINT
CHR$(7):CLS#2: GOTO 850
1890 CLS: LOCATE 2,2:PRINT"Ile fabryk za k"+
CHR$(254)+"adasz ? ";: GOSUB 1240
1900 IF x=0 THEN LOCATE #2,3,4:PRINT#2,0 :
GOTO 850
1910 IF x>INT(kon/cinw) THEN x=INT(kon/cinw)
1920 IF x+fab>10 THEN x=10-fab
1930 fbud=fbud+x:LOCATE #2,3,4:PRINT#2,x
1940 FOR i=fab+1 TO fab+x:b(i-1)=1:NEXT
1950 fab=fab+x:IF fab=10 THEN GOSUB 1970
1960 kon=kon-x*cinw:GOTO 710
1970 CLS: LOCATE 4,3:PEN 2:PRINT"LIMIT FABRYK
WYCZERPANY !":PEN 1
1980 FOR i=1 TO 1500:NEXT:PRINT CHR$(7)
1990 RETURN
2000 '
2010 ' KREDYT
2020 '
2030 BORDER 3:CLS#2: LOCATE #2,2,2:PRINT
#2,"Kredyt"
2040 IF ilkred>0 OR czas>18 THEN PRINT
CHR$(7):CLS#2: GOTO 850
2050 ilkred=1
2060 CLS:LOCATE 2,2:PRINT"Ile $ chcesz po"+CHR
$(252)+"yczy"+CHR$(246)+" ? ";: GOSUB 1240
2070 IF x=0 THEN LOCATE
#2,4,4:PRINT#2,0:ilkred=0: GOTO 850
2080 IF x>INT(fab*cinw/2) THEN
x=INT(fab*cinw/2)
2090 kred=x:czaskr=1:kon=kon+kred: LOCATE
#2,2,4:PRINT#2,x
2100 GOTO 710
2110 '
2120 'GIELDA
2130 '
2140 CLS#2: LOCATE #2,2,2:PRINT#2,"Gie"+
CHR$(254)+"da":BORDER 5
2150 CLS: GOSUB 2340: LOCATE 10,1:PRINT" KUPNO
AKCJI - <";:PEN 2:PRINT"K";:PEN 1:PRINT">"
2160 LOCATE 7,2:PRINT"SPRZEDA"+ CHR$(253)+"
AKCJI - <";:PEN 2:PRINT"S";:PEN 1:PRINT">"
2170 LOCATE 2,3:PRINT"POWR"+ CHR$(245)+"T -
<DOW. INNY KLAWISZ>"
2180 CLEAR INPUT
2190 a$=INKEY$: IF a$="" THEN 2190
2200 IF a$="k" OR a$="K" THEN 2230
2210 IF a$="s" OR a$="S" THEN 2420
2220 GOTO 850
2230 'ZAKUP AKCJI
2240 CLS#2:LOCATE#2,2,1:PRINT#2,"Zakup":LOCATE
#2,2,3:PRINT#2,"Akcji":BORDER 20
2250 IF ceakc>kon THEN PRINT CHR$(7):CLS #2:
GOTO 850
2260 CLS:GOSUB 2340:LOCATE 7,2
2270 PRINT"Ile akcji kupujesz ? ";: GOSUB 1240
2280 IF x>INT(kon/ceakc) THEN x=INT(kon/ceakc)
2290 LOCATE #2,3,5:PRINT#2,x
2300 IF x=0 THEN 850
2310 ilakc=ilakc+x
2320 kon=kon-ceakc*x
2330 GOTO 710
2340 PEN 2: LOCATE 1,4:PRINT USING "AKTUALNA
CENA AKCJI #######$";ceakc
2350 LOCATE 7,5: IF ilakc<1 OR ilakc>4 THEN
PRINT USING "POSIADASZ #### AKCJI";ilakc:ELSE
GOSUB 2370
2360 PEN 1:RETURN
2370 IF ilakc=1 THEN PRINT"POSIADASZ 1 AKCJ"+
CHR$(243):ELSE PRINT USING"POSIADASZ ####
AKCJE";ilakc 2380 RETURN
2390 '
2400 'SPRZEDAZ AKCJI
2410 '
2420 CLS#2: LOCATE #2,1,1:PRINT#2,"Sprzeda"+
CHR$(252): LOCATE #2,2,3:PRINT
#2,"Akcji":BORDER 16
2430 IF ilakc<=0 THEN PRINT CHRf(7):CLS#2:
GOTO 850
2440 CLS: GOSUB 2340:LOCATE 3,2:PRINT"Ile
akcji sprzedajesz ? ";: GOSUB 1240
2450 IF x=0 THEN LOCATE #2,3,5:PRINT#2,x:
GOTO 850
2460 IF x>ilakc THEN x=ilakc
2470 LOCATE #2,3,5:PRINT#2,x
2480 ilakc=ilakc-x
2490 kon=kon+ceakc*x
2500 GOTO 710
2510 '
2520 '
2530 ' KALKULATOR
2540 ON ERROR GOTO 2880
2550 BORDER 7:CLS:CLS#2: LOCATE
#2,2,2:PRINT#2,"Kalku ": LOCATE
#2,2,4:PRINT#2,"lator"
2560 LOCATE 10,4:PEN 2:PRINT"MA"+ CHR$(255)+"Y
KALKULATOR"
2570 PRINT TAB(2);"DNIE LICZBY CA"+
CHR$(255)+"KOWITE >0"
2580 PRINT TAB(2);"DZIA"+ CHR$(255)+"ANIA
+-*/s KONIEC <K>"
2590 PEN 1:WINDOW 10,24,20,22:PAPER 0:CLS
2600 x$="":y$="":INPUT a$: IF a$="k" OR a$="K"
THEN 2860
2610 l=LEN(a$)
2620 IF l=0 THEN 2600
2630 i=0
2640 i=i+1
2650 z$=MID$(a$,i,1)
2660 z=ASC(z$)
2670 IF z=32 THEN 2640
2680 x$=x$+z$
2690 IF z<48 OR z>57 THEN 2800
2700 GOTO 2640
2710 i=i+1:IF i=l+1 THEN 2780
2720 t$=MID$(a$,i,1):IF t$="" THEN y$="0":GOTO
2780
2730 t=ASC(t$)
2740 IF t=32 THEN 2710
2750 y$=y$+t$
2760 IF t<48 OR t>57 THEN 2870
2770 GOTO 2710
2780 x=VAL(x$):y=VAL(y$)
2790 IF y=0 AND z=47 THEN PRINT"Dzielenie
przez zero !": GOTO 2600:ELSE 2810
2800 IF NOT(z=42 OR z=43 OR z=45 OR z=47
OR z=94) THEN PRINT"Dzia"+ CHR$(254)+"anie
niedozwolone !": GOTO 2600:ELSE 2710
2810 IF z=42 THEN w=INT(x*y):PRINT w: GOTO
2600
2820 IF z=43 THEN w=INT(x+y):PRINT w: GOTO
2600
2830 IF z=45 THEN w=INT(x-y):PRINT w: GOTO
2600
2840 IF z=47 THEN w=INT(x/y):PRINT w: GOTO
2600
2850 IF z=94 THEN w=INT(x^y):PRINT w: GOTO
2600
2860 WINDOW 1,32,20,25:PEN 1:PAPER 3:CLS
:LOCATE#2,1,5:PRINT#2,w:w=0:ON ERROR GOTO
0:GOTO 850
2870 PRINT"Dzia"+ CHR$(254)+"anie niedozwolone
'": GOTO 2600
2880 PRINT"Dzia"+ CHR$(254)+"anie niedozwolone
!*:RESUME 2600
2890 WINDOW 1,40,1,25:PAPER 1:PEN 2:CLS
2900 '
2910 ' UPLYW CZASU
2920 '
2930 ' Sygnal dzwiekowy
2940 CLS#2:LOCATE#2,3,3:PRINT#2,"Czas"
2950 RANDOMIZE TIME
2960 czas=czas+1
2970 IF czas>74 THEN 3350
2980 FOR a=0 TO fabcz-1
2990 ipro=ipro+a(a,3)
3000 a(a,3)=a(a,2)
3010 a(a,2)=a(a,1)
3020 a(a,1)=0
3030 NEXT
3040 IF fbud>0 THEN 3050 ELSE 3100
3050 FOR a=fabcz TO fab-1
3060 b(a)=b(a)+1
3070 IF b(a)>6 THEN fabcz=fabcz+1
3080 IF b(a)>6 THEN fbud=fbud-1
3090 NEXT
3100 ccze=FNg(ccze):ctra=FNg(ctra)
3110 cpro=FNg(cpro):cinw=FNg(cinw)
3120 kon=kon+INT(kon/500)
3130 kon=kon-1000*fabcz
3140 IF kred>0 THEN GOSUB 3180
3150 IF kon<0 THEN 3240
3160 ceakc=ceakc-
INT(RND*ceakc)+INT(RND*1.2*ceakc)
3170 GOTO 710
3180 czaskr=czaskr+1
3190 IF czaskr>6 THEN 3220
3200 kred=kred+INT(kred*0.05)
3210 RETURN
3220 kon=kon-kred:kred=0:czaskr=7
3230 RETURN
3240 '
3250 'BANKRUCTWO
3260 '
3270 MODE 0:WINDOW 1,20,1,25
3280 INK 0,6:INK 1,0:INK 2,25:PAPER 0:PEN
1:CLS
3290 LOCATE 1,12:PRINT"B A N K R U C T W O!"
3300 LOCATE 4,15:PEN 2:PRINT"Chyba":
LOCATE 7,17:PRINT"stracisz": LOCATE
10,19:PRINT"prac"+ CHR$(242)+"..."
3310 FOR t=1 TO 100:SOUND 4,(79+t),5:NEXT
3320 FOR i=1 TO 2000:NEXT
3330  GOTO 3350
3340 czas=czas+1: GOTO 3350
3350 :MODE 1:WINDOW 1,40,1,25
3360 INK 0,4:INK 1,25:INK 2,6:INK 3,13
3370 BORDER 4
3380 PAPER 0:PEN 1:CLS: LOCATE 11,2
3390 PRINT"K O N I E C  G R Y  !"
3400 LOCATE 10,10:PRINT"Czas gry ";:PEN
2:PRINT czas-1;:PEN 1:PRINT" mies."
3410 LOCATE 10,13:PRINT"Zdoby"+CHR$(254)+"e"+
CHR$(248)+"  ";:PEN 2:PRINT kon;:PEN 1:PRINT"
$"
3420 PEN 3: LOCATE 10,23:PRINT"Ponowna gra -
klawisz <P>"
3430 LOCATE 10,25:PRINT"BASIC - dowolny inny
klawisz"
3440 CLEAR INPUT
3450 d$=INKEY$: IF d$="" THEN 3450
3460 IF d$="p" OR d$="P" THEN LOCATE 12,18:PEN
2:PRINT"S T A R T": RUN 440 ELSE CALL 0
3470 SYMBOL AFTER 237
3480 SYMBOL 240,0,0,&X1111000,12,&X1111100,&X1
1001100,&X1110110,12
3490 SYMBOL 241,24, &X111100,&X1100110,&X11001
10,&X1111110,&X1100110,12
3500 SYMBOL 242,0,0,&X111100,&X1100110,&X11111
10,&X1100000,&X111100,12
3510 SYMBOL 243,&X11111110,&X1100010,&X1101000
,&X1111000,&X1101000,&X1100010,&X11111110,12
3520 SYMBOL 244,6,12,&X111100,&X1100110,&X1100
110,&X1100110,&X111100,0
3530 SYMBOL 245,&X111011,&X1111100,&X11000110,
&X11000110,&X11000110,&X1101100,&X111000,0
3540 SYMBOL 246,6,12,&X111100,&X1100110,&X1100
000,&X1100110,&X111100,0
3550 SYMBOL 247,&X111101,&X1100110,&X11001100,
&X11000000,&X11000000,&X1100110,&X111100,0
3560 SYMBOL 248,6,12,60,&X1100000,60,6,
&X1111100,0
3570 SYMBOL 249,&X111101,&X1100110,&X1101000,6
0,6,&X1100110,60,0
3580 SYMBOL 250,6,12,&X1111110,&X1001100,24,4
8,&X1111110,0
3590 SYMBOL 251,14,&X11111110,&X11010110,&X100
01100,24,&X110010,&X1100110,255
3600 SYMBOL 252,12,0,127,&X1001100,24,48,127,0
3610 SYMBOL 253,255,&X11000110,12,&X1111110,48
,&X1100110,255,0
3620 SYMBOL 254,124,48,48,48,48,54,28
3630 SYMBOL 255,&X11110000,&X1100000,&X1101000
,&X1110000,&X11100010,&X1100110,25
3640 SYMBOL 238,6,12,&X11011000,&X1100110,&X11
00110,&X1100110,&X1100110
3650 SYMBOL 239,6,12,102,118,126,110,102
3660 RETURN
```

# ALIE




1 BAZA OPERACYJNA
28 ZBROJOWNIA
78 CENTRUM KONTROLI
174 GENERATOR
177-182 BLOK MEDYCZNY
248 POMIESZCZENIE KRÓLOWEJ

RYS: M. STAWICKI

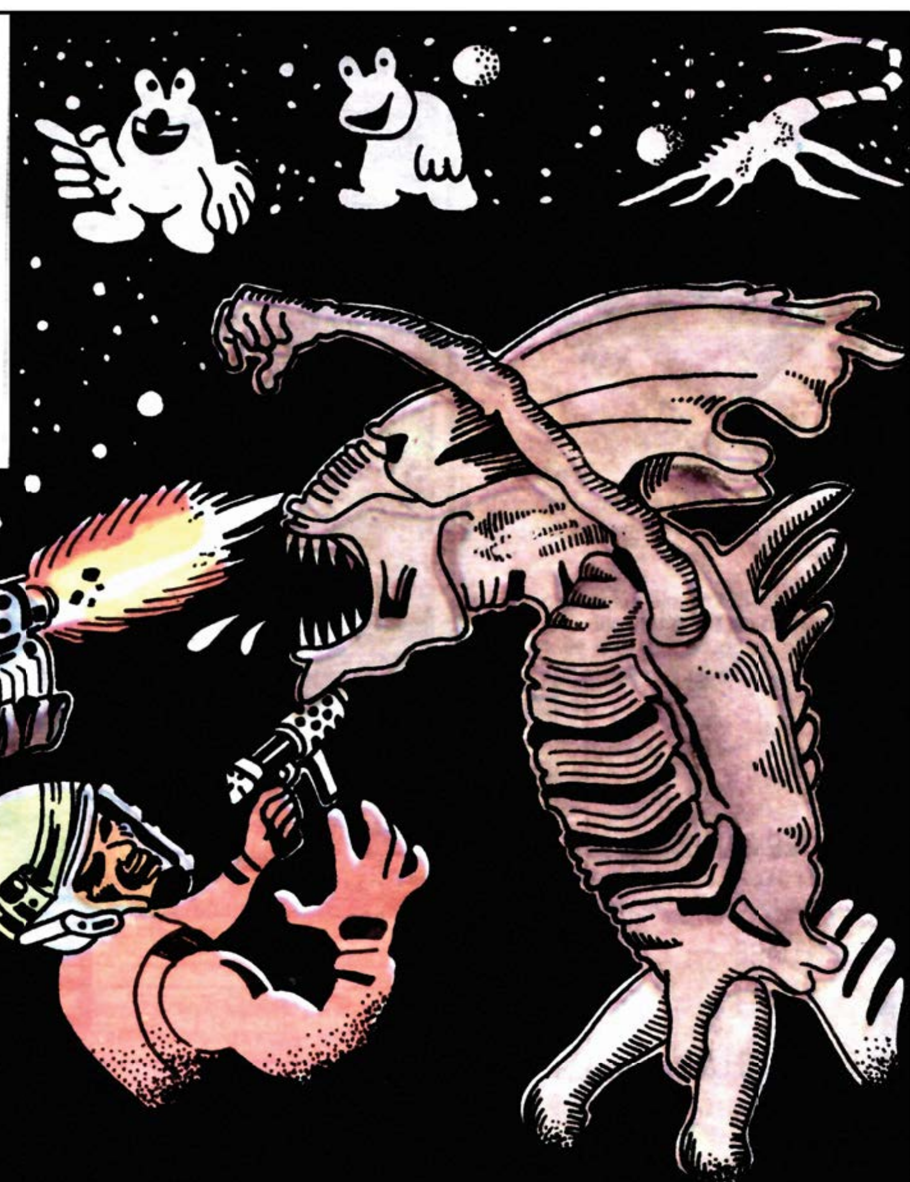
# ALIENS

Każdy chyba oglądał głośny horror SF „ALIENS" (Obcy — decydujące starcie) reżyserii James a Camerona, który jest znakomita kontynuacją filmu „ALIEN" (Obcy 8 pasażer Nostromo). Film „ALIENS" otrzymał wiele międzynarodowych nagród, w tym największą... Oskara, a konkretnie dwa Oskary: za efekty specjalne i dla odtwórczyni roli „Ripley" tj. dla Sigourney Weaver. Zgodnie z panującą aktualnie na zachodzie modą film doczekał się także swojego programu (gry komputerowej). Zarówno dla „ALIEN" jak i „ALIENS" były inspiracją dla programistów. Powstały więc gry o tych samych tytułach. My zajmiemy się grą ALIENS opartą na „drugiej" części „przygód" Ripley.

Sam program choć ma tak wiele wspólnego z filmem nie jest jego wierną kopią ale jest oparty na głównym (wg mnie) motywie filmu tj. poszukiwaniu królowej (Queen of Aliens). Zasadnicza różnica między filmem a grą polega na tym, że w grze królowej szuka sześcioosobowa ekspedycja, a nie Ripley w pojedynkę (jak w filmie). W misji bierze udział sześć osób spośród całej (rzeczywistej, wg filmu) załogi wysłanej na ratunek kolonizatorom. Sześcioosobowa ekspedycja, to:

1. RIPLEY — Flight Officer (stopień jeszcze z Nostromo) — obecnie cywil, ekspert od obcych, pilot.
2. BURKĘ — cywil, konsultant ds. kolonii pozaziemskich. Grupa komandosów (Space Marines)

# ENS



176
174 — 173
171 — 170
168 167
164 — 167
163
162
96 — 95 — 94
161 93 — 92 — 91
160 90 — 89 192 — 193 — 194 — 195 — 196 — 197 — 198
136 — 135 — 134 245 191 226 199 — 200 — 201 — 202
159 133 — 132 87 — 86 189 — 190 — 223 — 224 — 225 227 — 228 250 203
158 131 85 188 240 241 — 242 229 249 204
78 — 79 — 80 — 81 — 245 — 83 — 84 — 184 — 185 — 186 — 187 239 — 246 — 243 230 — 247 — 248 253 — 205 — 254 — 255
157 152 130 222 238 — 245 — 244 231 251 206
156 150 — 151 245 — 245 221 — 220 — 237 — 236 234 — 233 — 232 252 207
147 — 148 — 149 127 219 235 211 — 210 — 209 — 208
155 125 — 126 218 — 217 — 216 — 215 — 214 — 213 — 212
154 122 — 123 — 124
119 — 120 — 121
153
178 — 179
181 — 182
183

pod dowództwem kpt. Gormana.

3. GORMAN — Porucznik, marines, pilot, nawigator, elektronik (to jego pierwsza, nie symulowana akcja).
4. HICKS — Kapral, marines, ekspert ds. uzbrojenia, pilot (w filmie kreowany przez Michaela Biehn znanego już z filmu „TERMINATOR").
5. BISHOP — Szeregowy, marines android, elektronik — łącznościowiec, pilot.
6. VASQUEZ — Szeregowa, marines, zwiadowca, obsługa broni ciężkiej.

Tyle wprowadzenia, czas na grę.

Najważniejsza w całej grze jest amunicja, która zużywa się bardzo szybko, a uzupełnić ja można tylko w pomieszczeniu nr 28 (zbrojownia). A oto jeden z wielu (i na pewno skuteczny) sposób na pełny sukces misji: Proponuję podzielić całą grupę na dwa trzyosobowe oddziały. Grupa pierwsza bojowa i druga oczyszczająca.

Grupę bojowe pod dowództwem Ripley przeprowadź do pomieszczenia nr 78 ale ostrożnie i bez zabijania obcych po drodze. Najlepsza i najprostsza trasa to: z 1 do 4 potem w lewo, z 10 do 36 potem znowu w lewo do 69, potem prosto do 72, dalej w lewo do 107, do 96, w prawo do 161 i dalej już prosto do 78. Gdybyś w 78 napotkał obcego, to roznieś go na strzępy. Gdy cała trójka będzie już w komplecie, to jednym idź na zwiad prosto do 187, tam zabij obcego i skompletuj resztę grupy (tych z 78). Następnie spisując jedną z osób na straty (a może przeżyje?) wykonaj samobójczy marsz, niszcząc wszystkich obcych po drodze wg marszruty: 188-189-190-223-224-240-239-246-241-241- -227-228-229-230-247-248. Pozostałe dwie osoby muszą mieć pełne magazynki i „fuli" energii. Te dwie osoby doprowadź, już bezpiecznie do 247 ale uważaj, bo w 248 znajduje się królowa i „setki" stworów. Z zacięciem na twarzy roznieś królową i jej straże w drobny mak i oczyść pomieszczenie z jaj i kokonów. Pojawi się wtedy komunikat o możliwości powrotu do 1, jest to sygnał dla grupy oczyszczającej. Pozostałymi trzema (druga grupa) wytnij „do nogi" wszystkie pozostałe przy życiu stwory (kokony też) i wracaj do pomieszczenia nr 1. Przeprowadź grupę bojową do 1, oczyszczając drogę z „zapomnianych" obcych. Gdy wszyscy żywi będą już w pomieszczeniu nr 1 pojawi się komunikat o pomyślnym zakończeniu całej misji.

**UWAGA!**

Oto kilka przydatnych wskazówek:
1. Celuj zawsze w głowę.
2. Po oczyszczeniu sektory przez grupę drugą „zapieczętuj" go strzelając w „kontakt" od drzwi.
3. Nie zabijaj bezbronnych dzieci, to Newt, mała dziewczynka, dostaniesz za nią 1000 pkt. Oczywiście za jej odnalezienie, a nie zastrzelenie. Nie
4. Nie próbuj na siłę otwierać zamkniętych drzwi, bo nie wyjdzie ci to na zdrowie.
5. Strzeż się też trupów.
6. Nie używaj broni w pomieszczeniu generatora. Powodzenia, to decydujące i chyba ostatnie starcie (?).

**Producent:** **Electric Dreams**
**Autor:** **Soft Machine**
**Komputer:** **Amstrad/Schneider, Commodore 64/128, ZX-Spectrum 48/ + / 128/ + 2/ + 3**




Na nasze dziesiąte notowanie napłynęło 3456 propozycji, głosowano na 153 tytuły gier.

| | | | ATARI | AMSTRAD | COMMODORE | SPECTRUM |
|---|---|---|---|---|---|---|
| 1 | WINTER GAMES | ↑ | x | x | x | x |
| 2 | DAN DARE | ↓ | | | | x |
| 3 | WIZARD'S LAIR | ↓ | | | x | x |
| 4 | TRAP DOOR | | | | x | x |
| 5 | BROADSIDES | ↑ | x | | x | |
| 6 | BOULDER DASH | ↑ | x | x | x | x |
| 7 | REVOLUTIONS | ↓ | | x | | x |
| 8 | SILENT SERVICE | ↑ | x | x | x | x |
| 9 | TAU CETI | ↓ | x | x | x | x |
| 10 | XENO | ! | | | x | x |

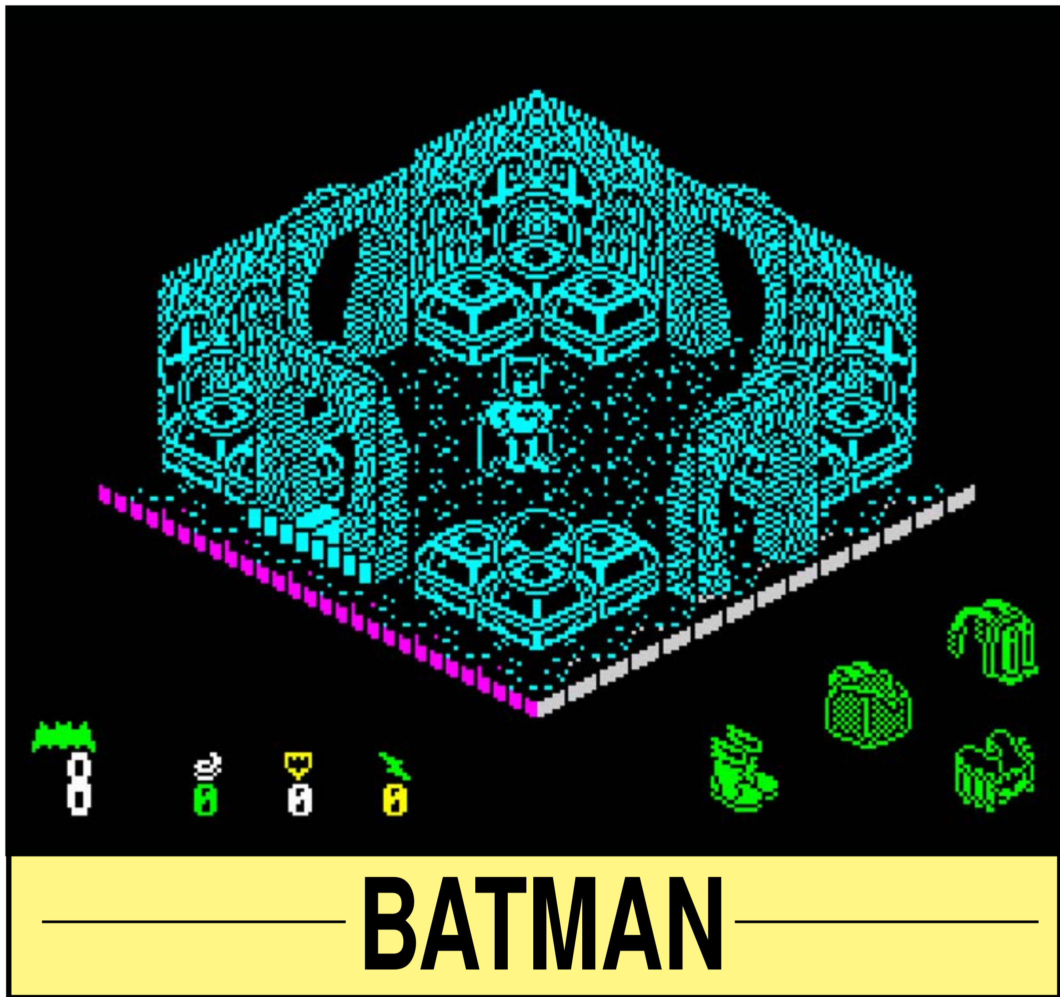
*Nagrody – zestawy programów komputerowych otrzymują: Magdalena Wieczorek i Jacek Czarko.*

*Sławek*




*Kopertę nadesłał Dariusz Dec ze Stalowej Woli, al. Wojska Polskiego 14*

# CO JEST GRANE



## BATMAN

Jak skończyć Batmana? To całkiem proste po przeczytaniu tych wskazówek. Wykonuj dokładnie to co mówi ten „opis" a twoją wyprawę uwieńczy sukces.

„Batman" polega na znalezieniu tzw. Batcraft'ów czyli części pojazdu którym Batman opuści podziemia stacji badawczej Kappa na Selen 7. Ponieważ Batman po przylocie na stacje został „rozbrojony" więc nie dysponuje żadnym ze swoich akcesoriów. Batman został uwięziony na najwyższym poziomie stacji Kappa. Aby podwyższyć swoją sprawność musi odnaleźć zabrane mu części garderoby tzn.:

Buty — pozwalają wykonywać krótki skok w każdą stronę,
Odrzutki — przyczepione do płaszcza wydłużają skok.
Antygraw — urządzenie, do sterowania podczas lotu lub skoku.
Torba-Plecak — może w nią zapakować wszystkie ruchome i względnie małe przedmioty ale tylko w obrębie jednego pomieszczenia.

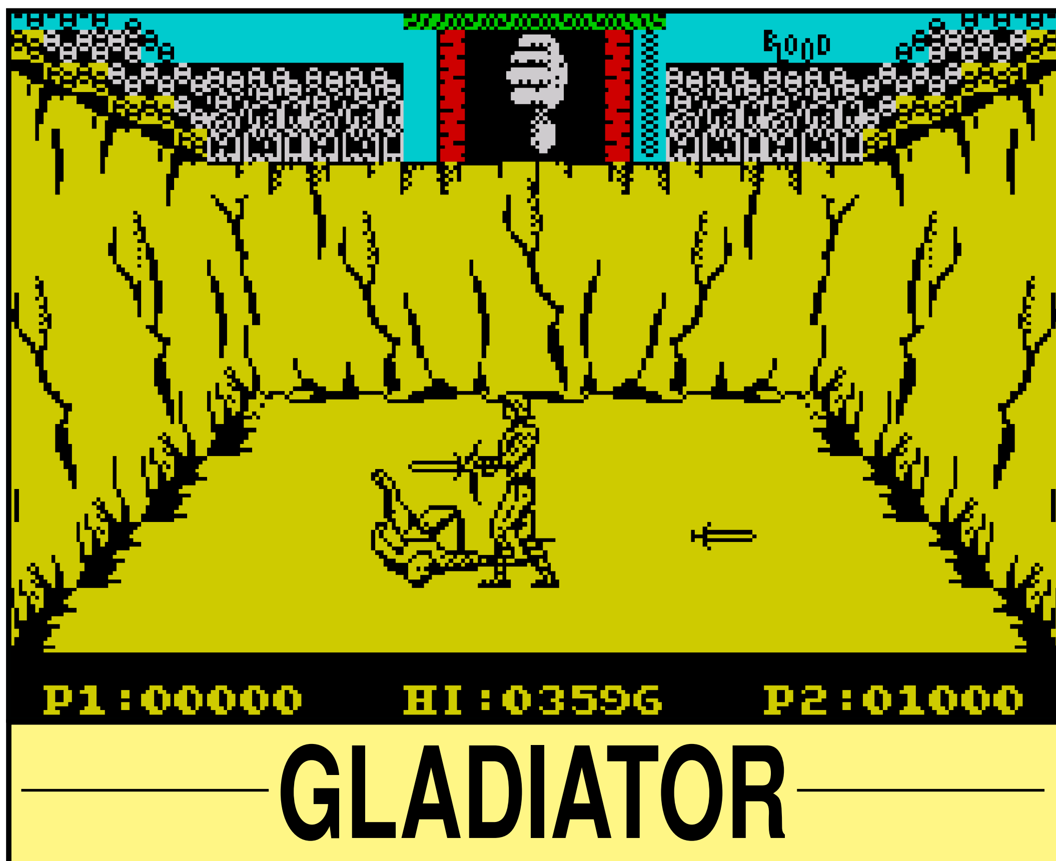
Po wydostaniu się spod kontroli straży, Batman rozpoczyna „zwiedzanie" stacji w poszukiwaniu „UNIFORMU" i Batcraft'ów. Uwaga, można spotkać wiele ruchomych postaci — wszystkie to wrogowie. Jedynie małe figurki podobne do bohatera pomagają w „misji" (tarcza lub chwilowo zwiększona prędkość może być „zbawienna").

A oto droga, którą Batman dąży do celu: W prawo, w prawo, w dót. weż przedmiot, w dół, w dół, weź przedmiot, w lewo, w lewo, w lewo, w lewo, w lewo, w górę, w lewo, w górę, weź przedmiot, w dół, w dół, w dół, w dół, w prawo, w dół, w górę przez sufit, weź przedmiot, w dół, w lewo, w dół, w dół, w prawo, w prawo, w prawo, w prawo, weź Batcraft'a, w górę, w lewo, w górę, w górę, w górę, w prawo, w prawo, w prawo, w dół, w lewo, w górę przez sufit, w prawo, weź Batcraft'a, w lewo, w dół przez podłogę, w prawo, w górę, w górę, w górę, w iewo, w górę, w górę, w lewo, weź Batcraft'a, w prawo, w górę, w lewo, w lewo. w lewo, w lewo, w dół, w lewo, w lewo, w lewo, w górę, w górę, w górę, w górę, w górę, w górę, w górę, w górę, w górę, weź Batcraft'a, w dół. w dół, w dół, w dół, w prawo, w dół, w prawo, w prawo, w prawo, weź Batcraft'a, w dół, w dół przez podłogę, w dół przez podłogę w prawo, w dół przez podłogę w prawo w górę. w lewo, w górę, w lewo, w górę, przez sufit, w prawo, w dół, weź Batcraft'a. w dói. w dół przez podłogę, w prawo, w dół, w dół przez podłogę, w dół przez podłogę, w dół, w dół, w lewo, w lewo, w dół, w dół, weź Batcraft'a, w górę, w górę, w prawo, w prawo, w górę przez sufit, w górę przez sufit, w górę przez sufit, w dół, w górę przez sufit, w górę przez sufit, w prawo, w prawo, w dół, w prawo, w dół, w dół, w lewo, w lewo, w lewo, w lewo, w lewo, w górę, w górę, w lewo, w lewo, w górę, w górę, w lewo, w lewo, w dół, w dół, w lewo, w górę przez sufit, w górę przez sufit, w dół, w dół, w górę przez sufit, w lewo, w lewo, skok do złożonego z Batcraft'ów pojazdu i koniec gry zarazem.

Powodzenia

*M.1*

**ZX Spectrum 48/+/128/+2/+3**
**Commodore 64/128.**




# GLADIATOR

**Urodziłem się** w Rzymie, w domu bogatego patrycjusza jako niewolnik. Bytem chłopcem leniwym a przy tym silnym i zręcznym więc mój pan nie mając ze mnie pożytku sprzedał mnie do szkoły gladiatorów. Dwa lata codziennych, wyczerpujących treningów pozwoliły mi wyjść na arenę i zmierzyć się z przeciwnikiem. Przeceniłem jednak swoje siły. Zostałem zraniony tak ciężko, że ledwo uszedłem z życiem. Od tego czasu moje treningi stały się dłuższe i staranniejsze. W końcu odniosłem moje pierwsze zwycięstwo. Za nim przyszło następne i nawet się nie obejrzałem, jak zostałem mistrzem Kampanii. Byłem ulubieńcem władcy Rzymu i ten obdarzył mnie wolnością. Nie mogłem jednak opuścić areny miałem zbyt wielu wrogów, którzy zniszczyli by mnie, gdybym wrócił do normalnego życia. Byłem więc skazany na walkę do końca, aż któryś z przeciwników okaże się lepszy. Dotychczas nikt nie zdołał mnie pokonać. Może Ty spróbujesz.

Do wyboru masz czterdzieści rodzajów broni — tarcze, miecze, włócznie, siekiery, widły, sieci, ciężarki i sztylety. Możesz wybrać trzy, w tym jeden ze sztyletów. Ja nie będę zmieniał broni, aż mnie pokonasz. Lecz nie łudź się — władca zawsze ułaskawi swojego ulubieńca wystawiając palec w górę. Natomiast Ty zostaniesz zawsze skazany na śmierć.

Podczas walki możesz rzucać bronią oraz podnosić ją z ziemi. Walczymy tak długo, aż jeden z nas powali drugiego na ziemię zagrażając jego życiu. Życzę Ci powodzenia lecz nie wierzę, abyś zdołał zająć moje miejsce.

Gra „Gladiator” firmy Domark jest prawdziwą nowością wśród gier o tematyce walki. Grafika i oprawa dźwiękowa jest bardzo staranna i dopracowana. Walczyć można z komputerem jak i z drugiem człowiekiem. Ta druga walka toczy się w lasku, ukryta przed wzrokiem ciekawskich jako że walki są w Rzymie zabronione.

Podczas gry używać można joysticka lub klawiatury. Poruszanie nie jest skomplikowane, lecz istnieje kilka dodatkowych ciosów (rzucanie, podnoszenie, rozpostarcie sieci) wykonywanych po dwukrotnym wciśnięciu przycisku FIRE. Ruchów tych jest sześć i podobno dla odkrywcy wszystkich producenci przewidzieli nagrodę.

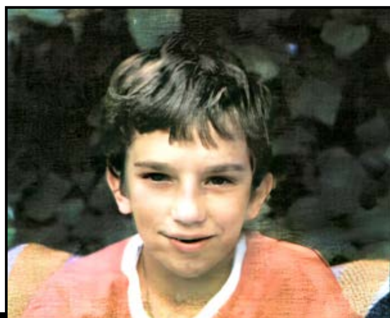
*(mp)*

# KRÓL I KRÓLOWA GIER



**Magdalena Wieczorek,** lat 14, uczennica VIII klasy Szkoły Podstawowej nr 47 we Wrocławiu.

Korzysta z komputera w klubie osiedlowym. Dlatego też jej ulubionym komputerem jest ZX Spectrum, zaś wymarzony to Commodore Amiga. Ulubiony program: Art Studio. Zainteresowania. Plany na przyszłość: muzyka, muzyka i jeszcze raz muzyka.



**Jacek Czarko,** lat 14, ósmoklasista, również z Wrocławia.

Korzysta z komputera w szkole, marzy o własnym Schneider CPC 464. Ulubiony program: WAR Interesuje się: dobrą muzyką współczesną i sportem — nie dla rekordów, lecz w celu rekreacji. Plany na przyszłość: mieć twórczą pracę.

*(b)*

# S.O.S.

Mam wielki kłopot. Poszukuję opisów i informacji na temat gier „Pitfall” i „BC'squestfortires”. Wtej drugiej szczególnie chodzi mi o trzecią przeszkodę po przeszkodach z żółwiami i następną z ptakiem.

Marek Szczodrak
58-561 Wojcieszyce 119A

Poszukuję dokładnego opisu gry „Robin of Sherwood” w wersji na mikrokomputer ZX Spectrum+ . Pomóżcie.

Robert Cieciera
ul. Grochowska 334
03-838 Warszawa

Pomocy! Nie wiem jak bawić się grami: „Pyjamara-ma”, „Rocky Horror”, „Eiffel Tower”, „Mancopter”, „The Fall Guy” na mikrokomputer Commodore 64,

Marcin Śliwa
ul. Mysłowicka 36
41-250 Czeladź

Proszę o opisy i czynności w grach „Cetewlaze” i „Pyjamarama”. Mam z nimi wielkie trudności. W zamian służę opisami do gier „Alien 8”, „Tomahawk”, „Sir Fred”, „Strike Force Cobra”, „Silent Service”.

Paweł Nowodziński
ul. Oleńki 6 m 9
42-200 Częstochowa

Kto mógłby mi odpowiedzieć na pytanie: jak należy grać na planszy B w grze „Boulder Dash II” w wersji na mikrokomputer Atari? Będę mu bardzo wdzięczny za odpowiedź. Liczę na Waszą pomoc.

Adam Sobiechart
ul. Żelazowej Woli 17/18
20-853 Lublin

Nie mogę sobie poradzić z grami: „Cave Lord”, „War Game's” i „New York City”. Mam Atari 65 XE. Bez waszej pomocy nie dam sobie rady.

Bartosz Janik,
ul. Energetyków 18/5
74-100 Gryfino

Usilnie poszukuję nieśmiertelności i innych ułatwień w programie „Dynamite Dan”. Pomóżcie! W wersji na ZX Spectrum.

Krzysztof Dęga
ul. Bydgoska 61 c/3
64-920 Piła

Bardzo proszę o informacje jak w grze Miner 2049'er przejść trzecią planszę. Nie wiem jak zamalować półkę, nad którą jest nr 3 i żelazko.

Bartek Danek
ul. Al. 1000-lecia 47/13
33-380 Krynica

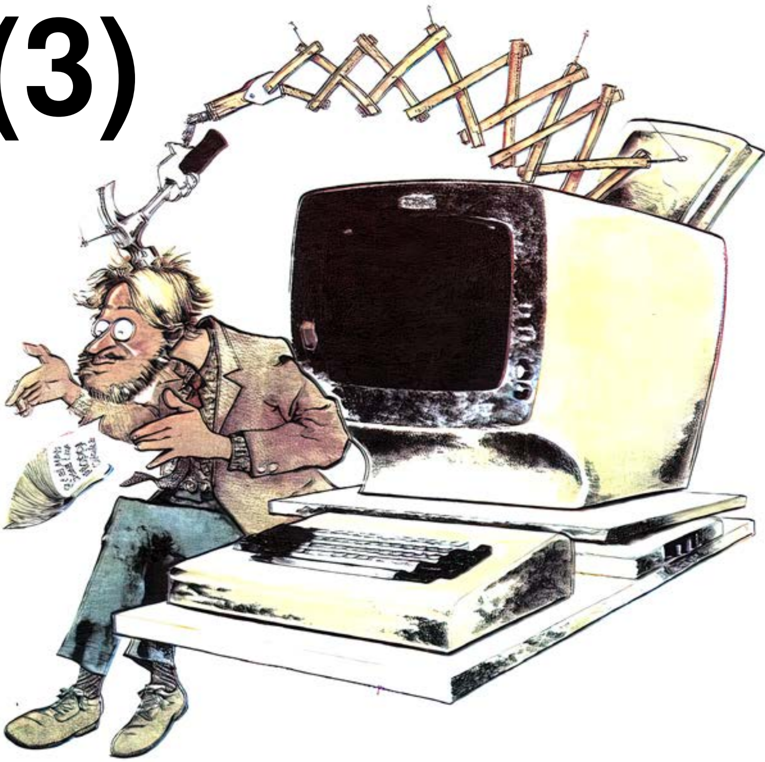
Mam kłopot, w wersji gry „Chequered Flag” na ZX Spectrum, nie wiem w jaki sposób zjechać do boksu naprawczego.

Maciek Poletyło
ul. Środkowa 5/4
20-015 Lublin


http://www.t2e.pl/reduksy

# NASTĘPNY KROK

# (3)



# INTERPRETER KOMPILATOR ASSEMBLER

**Zanim przejdziemy do asemblera. który będzie dziś naszym głównym tematem, jeszcze kilka słów o wyborze translatora..**

Porównując różne programy tego typu należy brac pod uwagę szybkość działania, ilość zajmowane) pamięci, ale także łatwość korzystania. Dalej, ważne jest, czy translator daje dobre, szczegółowe diagnostyki błędów, czy też pisze tylko „syntax error" i użytkownik musi głowić się sam na czym polega błąd. Warto też wziąć pod uwagę dodatkowe udogodnienia, takie jak np. możliwość śledzenia programu w trakcie wykonania, dołączanie procedur pisanych w innych lęzykach, dostęp do bibliotek podprogramów użytkowych, możliwość tworzenia własnych bibliotek. Decydującym czynnikiem będzie to, czy translator nie wymaga urządzeń, którymi nie dysponujemy, np. stacji dysków, i z drugiej strony, czy pozwoli wykorzystywać wszystkie urządzenia, które posiadamy np, drążek sterowy lub mysz.

Niezależnie od wszystkich rożnic i szczegółów technicznych języki wysokiego poziomu i ich translatory dość dobrze wykonują swoje zadanie, którym jest pośredniczenie między poziomem, na którym wygodnie jest pracować człowiekowi, a poziomem zer i jedynek języka wewnętrznego maszyny. Jednak nic za darmo. Ceną, którą trzeba zapłacie za taki model pracy jest zasłonięcie przed programistą części możliwości komputera. Przykłady: jeśli komputer ma gniazdo do podłączenia drążka sterowego, to na pewno procesor ma możliwość komunikowania się z tym gniazdem. Natomiast program np. w BASIC-u będzie miał tę możliwość tylko pod warunkiem, ze w repertuarze instrukcji rozpoznawanych przez translator znalazły się odpowiednie instrukcje wejścia/wyjścia. Rozmieszczanie programu i zmiennych w pamięci operacyjnej odbywa się automatycznie, przez co tracimy bezpośredni dostęp do tej pamięci*) Procesor może operować na pojedynczych bitach, zaś w wielu językach programowania możemy takie operacje wykonywać tylko za pomocą specjalnych sztuczek, lub nawet wcale nie możemy, itd

Wróćmy teraz na chwilę do historii. Przejście od pisania programu jako ciągu zer i jedynek do języków wysokiego poziomu nie nastąpiło jednym skokiem. Etapów pośrednich było kilka, a jednym z nich był właśnie assembler — narzędzie z jednaj strony na tyle wygodne, ze może być zaakceptowane przez człowieka, z drugiej strony dające pełny dostęp do wszystkich możliwości wbudowanych w maszynę przez jej konstruktorów.

Dzieje się to dlatego, że assembler jest dużo bliżej języka wewnętrznego niż języki programowania wysokiego poziomu. W zasadzie jeden rozkaz asemblera jest przekładany na jeden rozkaz procesora.

Skoro użycie asemblera „odsłania" nam procesor, to warto najpierw przyjrzeć się temu urządzeniu nieco dokładniej. Oczywiście wiele szczegółów technicznych musimy pominąć, aby otrzymać model odpowiednio ogólny. Wewnątrz procesora znajdziemy zwykle tzw. arytmometr, czyli jednostkę wykonującą wszystkie operacje arytmetyczne i logiczne na ciągach bitów. Żeby arytmometr miał skąd pobierać dane i gdzie umieszczać wyniki operacji, procesor zawiera rejestry, czyli urządzenia pozwalające zapisywać ciągi bitów — zwykle takiej samej długości jak długość komórki pamięci. Dwie grupy rozkazów procesora już można przewidzieć: dodanie, odjęcie itd. zawartości dwóch rejestrów i umieszczenie wyniku w rejestrze, oraz druga grupa, wszelkiego rodzaju przesłania danych między komórkami pamięci a rejestrami, np. wpisz zawartość komórki pamięci o podanym adresie do rejestru A. Odwołajmy się tu do języków programowania, prosta instrukcja A=B + C będzie zrealizowana jako następujący ciąg instrukcji procesora: załadowanie wartości zmiennych B i C z odpowiednich komórek pamięci do rejestrów procesora, dodanie wartości rejestrów i zapamiętanie, także w rejestrze, wyniku przepisanie wyniku z rejestru do odpowiedniej komórki PaO**). Wydaje się to być dosyć pracochłonne, ale idzie sprawnie, bo procesor wykonuje przynajmniej kilkaset tysięcy takich elementarnych operacji w ciągu sekundy.

Wróćmy do rozkazów. Wspominałem, że są one zapisane w kolejnych komórkach pamięci, i — w takiej samej kolejności jak były zapisane — pobierane do wykonania przez procesor. Jednak w programach mogą występować instrukcie warunkowe i wtedy część kodu może być pomijana. Zrealizujemy to dzięki następnej grupie — instrukcjom skoku. Ich wykonanie to nic innego jak „samoprzestawienie się procesora na pobieranie rozkazów do wykonania z innego miejsca PaO. Oczywiście po takim przestawieniu dalej obowiązuje pobieranie rozkazów po kolei — aż do napotkania na- stępnej instrukcji skoku

Dodajmy jeszcze instrukcje przesyłania zawartości rejestrów z (do urządzeń wejścia) wyjścia i będziemy mieli bardzo ogólny i znacznie uproszczony obraz procesora***) — a więc równocześnie obraz tego. co można zrobić w asemblerze. Postawmy się teraz w roli pionierów informatyki i spróbujmy pomyśleć, jak nie tracąc żadnych możliwości maszyny, ułatwić sobie życie.

Na pewno warto nadać wszystkim rozkazom procesora krótkie łatwe do zapamiętania nazwy, których będziemy używać zamiast zero-jedynkowych kodów. To jednak nie wszystko. Większość rozkazów wymaga podania jako dodatkowej informacji adresu fizycznego komórki pamięci, której rozkaz ma dotyczyć. Bardzo przyda się możliwość nadawania komórkom nazw i używania później tych nazw w miejscach, gdzie musi wystąpić adres. (Właśnie z tej możliwość zastępowania adresów symbolami wzięła się inna nazwa assemblera — „język adresów symbolicznych"). Także rejestry, które w procesorze mają binarne identyfikatory, otrzymują nazwy. Ważnym udogodnieniem jest możliwość podawania w programach liczb w systemach innych niż dwójkowy — najczęściej dziesiętnie, ósemkowo, szesnastkowo.

Dla opisanego języka potrzebny jest translator, który przełoży napisane w nim programy na język wewnętrzny maszyny. Jego działanie jest stosunkowo proste. Nazwy rozkazów trzeba zastąpić odpowiednimi kodami, zamiast nazw komórek pamięci wpisać odpowiednie adresy fizyczne, wszystkie liczby przeliczyć na wartości dwójkowe. Taki translator nazywany jest zwykle asemblerem — to nie pomyłka, to samo słowo oznacza dwie różne rzeczy: język i translator tego języka.

Nie jest również pomyłką wyliczenie wśród funkcji translatora tylko „mechanicznego" przekodowania zapisu instrukcji na binarne kody. gdyż wszystkie pozostałe problemy programista musi rozwiązać sam. Mam tu na myśli np. rozmieszczenie w pamięci danych i programu, czy dbałość o sens wykonywanych operacji. BASIC nie pozwoli dodać zmiennej liczbowej do zmiennej tekstowej, w assemblerze nie ma zmiennych, są tylko komórki pamięci, a w nich bity. Sprzęt daje możliwość dodania do siebie zawartości dowolnych dwóch komórek pamięci, assembler ma nam udostępnić wszystkie możliwości sprzętu, więc do naszej decyzji zostaje czy będziemy dodawać do siebie komórki, w których poprzednio zapisaliśmy teksty lub rozkazy dla procesora (zdarzają się sytuacje, w których wykonanie takich operacji jest celowe).

Programowanie w assemblerze jest trudniejsze niż w językach wysokiego poziomu, wymaga głębszej znajomości działania sprzętu, jednak w wielu zastosowaniach bywa potrzebne, lub nawet niezbędne. Dodatkową zaletą assemblera jest możliwość stworzenia dla fragmentów programu dużo efektywniejszego kodu wynikowego niż wygenerowany przez translator języka wysokiego poziomu. Często tryb postępowania jest następujący: do programu pisanego w języku wysokiego poziomu dołącza się podprogramy w assemblerze. Jest to nic innego jak wspomniany poprzednio dobór języka programowania do rozwiązywanego problemu — tutaj w ramach tego samego zadania.

Powszechnie uważa się również, ze umiejętność programowania w assemblerze to jakby wyższy stopień wtajemniczenia. Programiści znający chociaż jeden assembler lepiej rozumieją sposób, w jaki komputer wykonuje program napisany w języku wysokiego poziomu, a to pozwala im pisać lepsze programy,

*Andrzej Pilaszek*

***) większość interpreterów BASIC--a na mikrokomputery pozwala odwoływać się bezpośrednio do PAO instrukcjami PEEK i POKE. Bezpieczne są jednak tylko odwołania do obszarów, których zawartość jest dokładnie opisana w instrukcji, np. pamięć obrazu.**

****) oczywiście wiele zależy od konkretnego procesora. Np. można spotkać rozkaz pozwalający dodać zapisaną w rejestrze wartość bezpośrednio do zawartości komórki pamięci, bez pobierania tej drugiej do rejestru.**

*****) pominąłem m.in. istnienie grupy szczególnych wąsko specjalizowanych rejestrów..**


http://www.t2e.pl/reduksy

**Polanglia Ltd**

58 St Mary's Road, London W5 5EX
Tel.: 0-0441-840 1715 telex 946581
Konto: 70736805 BARCLAYS BANK
Ealing Bwy, London W5 (kod 20-27-48)
wyłączne przedst. na Polskę firmy

AMSTRAD

oferuje po **NAJNIŻSZYCH CENACH W EUROPIE**
Komputery **AMSTRAD PC 1512** (kompat. IBM)
już wraz z „Export Licence"
Drukarki AMSTRAD DMP 4000 (NLQ)
którymi Amstrad zdobywa również
1–8 miejsce na rynku drukarek
oraz NAJPOPULARNIEJSZE CPC 6128, PCW 8256
i 8512, Sinclair Spectrum Plus 2 (Amstrada)
drukarki STAR, itp.

Na zakupiony u nas sprzęt dostępny jest dodatkowo
SERWIS GWARANCYJNY wykonywany przez znaną
firmę **REFLEKS, ul. Glogera 1, W-wa.**

**ATARI**
**ZX SPECTRUM**

**INSTRUKCJE ● OPISY ● LITERATURA**
**SZKOŁY I KLUBY—ZNIŻKA**
**KATALOGI — GRATIS**
**CO PIĄTY PROGRAM - GRATIS**
**WYSYŁKA NA CAŁY KRAJ**

**Wypożyczalnia programów**
**D.H. "Sezam" II p., g. 16.00—19.00**
**00-849 Warszawa UPT 66 skr. p. 14**

# NOWA RODZINA KOMPUTERÓW OSOBISTYCH FIRMY IBM

**Sześć lat temu IBM wyprodukował swój pierwszy komputer serii PC uruchamiając tym samym lawinę naśladowców i narzucając rynkowi komputerów swój standard. W tym roku „Big Blue" uderza raz jeszcze.**

Drugiego kwietnia 1987 r. IBM zaprezentował nowa rodzinę komputerów: „Systemy Osobiste /2".

## *PERSONAL SYSTEM/2*

Jakkolwiek komputery te są pewną kontynuacją myśli popularnych i u nas PC, jednak oprócz głównego systemu operacyjnego MS-DOS i procesorów Intela wszystko w nich jest nowe. Nowy jest standard video i rodzaj monitora, nowy standard dyskietek, zupełnie nowa architektura komputera, nowy rodzaj kart dodatkowych itd.

Rodzinę tworzy pięć komputerów:

**Model 25**

Najmniejszy z rodziny. Dwie stacje 3.5" dyskietek (720 kB każda) znajdują się w jednej obudowie z monitorem. Miejsce na dwie dodatkowe karty. Pamięć w wersji podstawowej 512 kB. Reszta wyposażenia jak w modelu 30.

**Model 30** (Clone killer):

Mniejszy i lżejszy od starych PC-etów. Jest z nimi kompatybilny. Główny procesor to Intel 8086 działający z częstotliwością zegara 8MHz bez stanów oczekiwania. System operacyjny to nowy DOS 3.3. Szyna danych jest 16 bitowa. Pamięć od 640 kB do 2.6 4 MB.

Na płycie głównej komputera jak w każdym komputerze „P-S/2" znajdują się:

— port szeregowy
— port równoległy
— kontroler dyskietek i dysku twardego
— gniazda monitora i myszy

Oprócz powyższych, Model 30 posiada trzy gniazda na dodatkowe karty kompatybilne ze standardem PC. Moc obliczeniowa jest uważana za dwa razy większą od starego XT.

Wersja podstawowa wyposażona jest w dwa 3.5" napędy dyskietek (720 kB pojemności każdy) czarnobiały monitor i klawiaturę i kosztuje ok. 5 tys. DM. Z jednym napędem 3.5" dyskietek i 20 MB dyskiem twardym cena wynosi ok. 6300 DM

**Model 50** (Standard)

Jest to podstawowy model nowej rodziny z nową mikrokanałową architekturą. Procesor to Intel 80286 z 10 MHz zegarem ze stanami oczekiwania. Prędkość komputera porównywana jest z XT 286. Pamięć komputera może być od 1 MB (DOS 3.3 korzysta tylko z 640 kB) do 7 MB. Oprócz tego dla tego komputera przeznaczony jest nowy system operacyjny tzw. „Operating System/2" - OS/2.

Wyposażenie podstawowe Modelu 50 jest analogiczne jak Modelu 30. Inny jest standard grafiki — Video Graphic Array (VGA) i wysokiej rozdzielczości analogowy monitor.

Model ten posiada trzy gniazda na dodatkowe karty niekompatybilne ze starymi.

Zestaw wyposażony w 3.5" napęd dyskietek i 20 MB dysk twaidy kosztuje ok. 10 tys. DM. Pojemność stacji dysków twardych można zwiększyć do 185 MB.

**Model 60** (nowy AT):

Charakteryzuje się zupełnie nowym wyglądem. Na stole stoi tylko klawiatura i monitor. Reszta komputera umieszczona jest w pionowo ustawianej na podłodze obudowie. Jest w niej miejsce na siedem dodatkowych kart nowej serii (16 bitowych) i napęd 5.25" dysku twardego o podwyższonej pojemności do 185 MB.

Moc obliczeniowa tego modelu ocenia się na 20% do 50% większą od PC-AT.

Procesor 80286 pracuje z 10 MHz zegarem bez stanów oczekiwania. Pamięć od 1 MB do 15 MB. Posiada 128 kB pamięci ROM. Twarde dyski o pojemności od 44 MB do 70 MB mają czas dostępu ok. 40 ms.

Głównym systemem operacyjnym ma być OS/2 zapowiadany na 1988 r.

Zestaw wyposażony w napęd 3.5" dyskietek (1.44 MB) i 44 MB dysk twardy kosztuje ok. 14 tys. DM a z 70 MB dyskiem twardym ok. 15.2 tys. DM.

**Model 80:**

Wygląd tego modelu jest analogiczny jak modelu 60, jednakże we wnętrzu kryje się już minikomputer o prędkości 25-krotnej większiej niż stare PC.

Procesor 80386 pracuje z 16 lub 20 MHz zegarem, w zależności od wersji. Pamięć od 2 MB do 20 MB (22 MB). Pojemność dysków twardych od 44 MB (115 MB) do 230 MB z czasem dostępu 28 ms. W obudowie są trzy 32 bitowe gniazda i cztery 16 bitowe gniazda kart dodatkowych. Wyposażenie podstawowe jak każdego modelu PS/2. Karta grafiki to VGA. Ceny w zależności od wersļi to 16.6 tys. DM (44 MB/16 MHz) i 24.1 tys. DM (115 MB/20MHz).c — rewolucji w komputeryzacji na razie nie widać. Kierunek

w jakim zmierza świat jest jasny dla wszystkich: komputer musi być mniejszy, tańszy i lepszy. Połączenie tych wszystkich cech w jednym modelu jest rzadko spotykane. Na ogół firmy skupiają się na jednej z cech, kosztem dwóch pozostałych.

## CO NOWEGO W ŚRODKU

Do konstrukcji systemu wprowadzono wiele ulepszeń. Oto wymienione niektóre z nich

— lepsze procesory
— zegary o wyższej częstotliwości
— przyspieszona współpraca z twardym dyskiem
— „szersze" magistrale adresowe i danych
— mikrokanałowa architektura

Ta ostatnia nowość oznacza wymianę danych między elementami komputera tylko przy pomocy układów DMA. Kanałów transmisji jest w tych komputerach 15 (w PC są tylko dwa), dla wykonywania czterech różnych rodzajów wymian danych. Umożliwia to wprowadzenie wielozadaniowości i architektury wieloprocesorowej (do 15 sztuk). Kanały DMA są 16 i 32 bitowe.

Komputery zbudowane są z elementów wielkiej skali integracji, a szczegóły techniczne nie są ujawniane co spowoduje olbrzymią trudność w produkcji kopii przez konkurencję.

Oprócz powyższych wszystkich zalet są też pewne wady. Nowy system jest niekompatybilny hardware'owo ze starym.

## NOWA GRAFIKA

Wprowadzone zostały trzy nowe standardy grafiki komoatybilne z CGA i EGA. Wymagają one nowych analogowych monitorów RGB. Producent proponuje cztery rodzaje monitorów (komputer sam rozpoznaje rodzai pod łączonego monitora).

**1. MCGA:** standard modelu 25 i 30. (Multi color graphics array) — 320x200 punktów; 256 kolorów z palety o 266000 kolorach (lub 64 odcień szarości); 640x480 punktów w trybie monochromatycznym

**2. VGA;** (Video Graphics Array Standard systemu (modele 50, 60 i 80). Matryca znaków 9x16 punktów w 43 wierszach i 30 kolumnach 320x200 punktów; 256 kolorów z pa lety o 266000 kolorach (64 odcienie szarości). 640x480 punktów przy 16 kolorach

3. Standard grafiki wysokiej rozdzielczości. 1024 x 768 punktów; 256 kolorów z palety o 266000 kolorach (pamięć karty ok. 1 MB)

## SYSTEMY OPERACYJNE

**System DOS 3.3**

System ma być bardziej „przyjacielski". Zawiera nowy interpreter BASIC-a. Nie zawiera programów linker i Exe2bin, które są opcjonalne.

Dodane są min. nowe funkcje Fastopen (o 15% szybsza), Call i Append. Również możliwe jest uruchomienie czterech łączy szeregowych o szybkości transmisji do 19200 bd.

**System OS/2**

Ma to być ten długo oczekiwany system operacyjny, wykorzystujący wszystkie zalety procesorów rodziny 80286. Tzn. umożliwia:

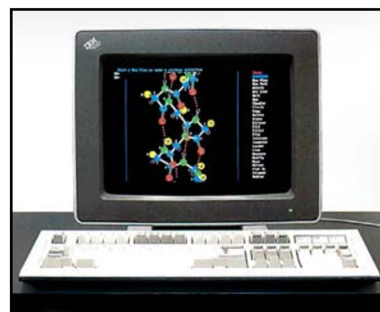
— prace w trybie zależnym
— do 16 MB bezpośredniego adresowania
— adresowania pamięci wirtualnej
— wielozadaniowość
— współpracę z twardym dyskiem o pojemności większej niż 32 MB.

System ma również działać na PC-AT.

*Krzysztof Czernek*


Model 50


Model 60


Model 80



Model 30

BIURO USŁUG KOMPUTEROWYCH
BONUS
● PROGRAMY
● LITERATURA
– ATARI 800 XL65, 130 XE/520 ST
– AMSTRAD 464, 664, 6128
– ZX SPECTRUM
– COMMODORE C64/C128
– IBM – opracowania literaturowe
04-111 Warszawa
ul. Grochowska 207
tel. 100-061 wewn. 244
w godz. 16.00–19 00
– rachunki dla instytucji
– informacje po nadesłaniu koperty zwrotnej.

VIDEO-BIT
ATARI ● SPECTRUM ● TIMEX
programy pocztą i na miejscu, rachunki. Tylko u nas co 10-ty program GRATIS' VIDEO-BTT ul. Małachowskiego 2. P--100 18-403 Łomża 5 D-140

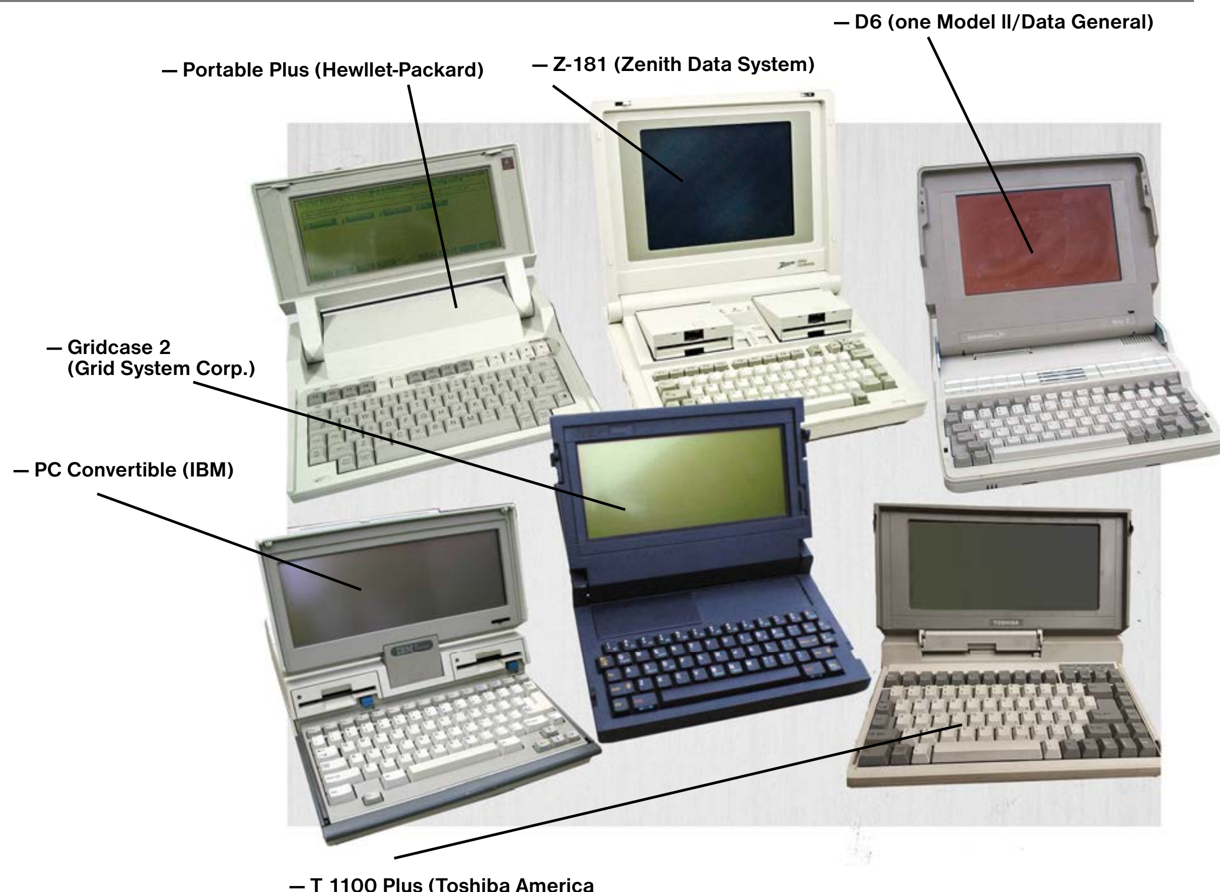
Naprawa yojsticków
Warszawa, Grochowska 144.

STUDIO ATARI " H O O Z Y "
Sopot. Boh. Monte Cassino 21/D
– wypożycza programy komputerowe
– udostępnia sprzęt
Napisz lub wstąp! Codziennie oprócz poniedziałków 12.00-20.00.
Gwarantujemy jakość i terminowość!
Tylko u nas najlepszy program graficzny na kasecie!

INTERFACE DO SPECTRUM
system Kempston .......... 6670,- zł
system Sinclair (dla dwóch joysticków) .......... 8200,- zł
Wysyłka natychmiastowa za zaliczeniem pocztowym
Roczna gwarancja
Dla instytucji rachunki płatne przelewem
Elektromechanika ul. Cegielniana 17. 32-410 DOBCZYCE

ATARI REWELACYJNE CENY — PONAD 1000 PROGRAMÓW IRATA SOFTWARE. SKR. POCZT. 160. 66-400 Gorzów Wielkopolski.
tel 249-58

hobby video computer
— nagrywa i wypożycza programy (ATARI, SPECTRUM)
— wypożycza komputery, odtwarzacze video, TV PAL
— oraz organizuje pokazy video dla świetlic, Klubów itp
— konserwacje urządzenia video i komputery
KATOWICE ul Rynek 12 S.D.H "ZENIT" III p.




# MAŁE JEST LEPSZE

**Producenci komputerów typu portable całymi latami próbowali zdobyć dla swych przenośnych modeli potencjalnych kupców reprezentujących — jak ocenia się — rynek około 9 miliardów dolarów.**

Nienajlepsze jednak dotąd ekrany, brak zgodności między poszczególnymi modelami i niechęć wielkich korporacji do zamawiania CD portable ograniczyła ich zasięg do użytkowników, dla których niezbędny był (i jest) word processor w walizce.

Według Andy'ego Czernka z Zenith Data teraz wszystko się zmieniło; również w przypadku tych urządzeń opcje wielkich producentów zbliżają się do upodobań nabywców. Wyprodukowane przez Zenith Data System ekrany z ciekłych kryształów posiadają jakość porównywalną z dotychczas stosowanymi monitorami kineskopowymi. Ponadto trwała i zajmująca niewiele miejsca stacja 3,5 calowych dysków powoduje, że urządzenie jest nieduże, lekkie i — co najważniejsze — zgodne z innymi komputerami tej walizkowej rodziny.

Żeby nie być gołosłownym — sprawą LCD portables zainteresował się IBM powołując zespół dla przygotowania swojego „kieszonkowego" modelu.

A cała sprawa rozpoczęła się stosunkowo niedawno. W 1984 roku CAta General Corp. pokazał na rynku pierwszy pełnoekranowy komputer przenośny z ekranem na ciekłych kryształach. Nazwane go DG/One. *Niestety — cząstki ciekłego kryształu skręcały się tylko o 90 stopni dając na ekranie światło spolaryzowane* — lapidarnie starał mi się wyjaśnić te problemy odpowiedzialny za konstrukcję DG/One Ron Pipe. *W efekcie* — kontynuował — *kontury obrazu były rozmazane a powierzchnia niewyraźna. Ponadto obraz był zniekształcony a współczynnik kontrastu ograniczony do około 2:1* (Obraz był tylko dwa razy jaśniejszy od tła i trudny do percepcji nawet przy maksymalnym wzmocnieniu światła).

— *W następnej generacji ekranów używanych w DG/One Model III, kąt skręcenia cząsteczek był większy i wynosił już 180 stopni* — wyjaśnia *Pipe Dało to w efekcie większą niż poprzednio różnicę między tłem a obrazem. Zwiększył się współczynnik kontrastu, który w tym modelu wynosił 6 1. Taki efekt w porównaniu z pierwszą wersją wywierał już wrażenie. Żadnego z tych modeli nie daje się jednak porównać z. Zenithem Z-181.*

— *Nasz najnowszy ekran LCD ma kontrast 12:1* — twierdzi nie bez dumy Czernek — *i zaczyna być porównywalny z monitorami katodowymi (kontrast okol 20 : 1). W modelu Zenitha cząsteczki skręcają się o 270 stopni: „superobrót" w znacznym stopniu odpowiedzialny jest za czystość obrazu i kontrast. Obraz pojawia się od razu* — wyjaśnia Czernek. *Nie ma żadnych odbić i refleksów po załączeniu i wyłączeniu ekranu. Dodatkową korzyścią poprawy kontrastu jest uzyskana możliwość szerokiego kąta widzenia.*

Żeby zobaczyć obraz na wcześniej szych modelach LCD portable należało patrzeć wprost na ekran a maksymalne odchylenie, przy którym obraz był jeszcze czytelny wynosiło ± 30°. Natomiast ekran LCD modelu Z-181 jest czytelny nawet przy kącie ± 70°. *Jest to szczególnie ważnie* — dodaje Czernek — *gdy obraz oglądany jest przez kilka osób.*

*Można zapytać dlaczego chcieliśmy osiągnąć takie wyniki* — pyta Andy i odpowiada natychmiast: *jest to po prostu dobry biznes. Przed komputerami typu LCD portable (mikrokomputer walizkowy z ekranem na ciekłych kryształach) otwiera się mnóstwo zastosowań; dodatkową zaletą jest ich zgodność z dotychczasowymi standardami PC.*

— *Nie mogę powiedzieć, że to my rozpoczęliśmy nowy standard, ale prawdą jest, że popieraliśmy wprowadzenie dyskietek 3,5 calowych* — twierdzi Ed Thomas, zajmujący się w IBM systemami operacyjnymi.

Kiedy IBM popiera cokolwiek — tak jak 3,5 calowy FDD w swoim nowym PC Convertible — trzeba by nie mieć oleju w głowie, żeby nie zrozumieć zarysowującego się trendu.

Następna zaletą LCD portable jest, że akceptują ten sam MS-DOS — kontrolny software.

— *Powoduje to, że są zgodne z innymi oraz z desktop PC* — twierdzi Murphy. *To stwarza możliwość zastosowania daleko poza życie ich jedynie jako walizkowe „inteligentne" maszyny do pisania.*

— *Największe zapotrzebowanie zgłaszają instytucje ubezpieczeniowe i uniwersytety* — wyjaśnia Ed Thomas. *Inne zastosowania zawierają możliwość użycia ich w bankach, działach sprzedaży, podczas różnego typu kontroli i w trakcie pracy badawczej.*

— *Póki co jednak potencjalni ich użytkownicy stoją po stronie komputerów osobistych* — przyznaje Murphy. *Trudno im się dziwić, że wybierają systemy i urządzenia powszechnie stosowane proponowane przez największe firmy. Ale do czasu.*

LCD portables pracują podobnie jak wszystkie PC zgodnie z IBM; wkładając dyskietkę systemową MS-DOS do stacji dysków załączamy komputer. Po chwili gotowy jest już do pracy. No, prawie gotowy...

Potrzebny jeszcze jest software, który można znaleźć lub zaadaptować na wiele różnych sposobów. Na przykład IBM sprzedaje go już na dyskietkach 3,5 calowych. Gried Corp. proponuje kopiowanie programów zawartych na 5,25 calowych dyskietkach poprzez ROM. Toshiba America oferuje 5,25 calową dyskietkę dołączoną do LCD portable w celu skopiowania programów na dysk 3,5 cala. Jeszcze inną metodę nazwano „portingiem". Używając Desktop PC przesyła się — przewodem z odpowiednią wtyczką poprzez „port" — wersję programu z dyskietki 5,25 cala na 3,5 calową w LCD portable (Komunikacyjny software plus przewód z wtyczką można dostać za 130 dolarów).

— *Komputery walizkowe bądą wkrótce jeszcze mniejsze, lżejsze i sprawniejsze*— stwierdził Ron Pipe.

A jeżeli chodzi o ekran!

— Będzie kolorowy — dodał — *najdalej za dwa, trzy lata.*

*William J. Hawkins*
*(opr. F.P.)*

UŻYTKOWNICY ATARI XL/XE
ATAREX oferuje TANIO ! duży wybór programów do komputerów ATARI
na taśmach kasetowych oraz dyskietkach.
Szczegółowych informacji po załączeniu znaczka udziela:
ul. 22 Lipca 17
62-300 WRZEŚNIA
ATAREX
ul. 20 Października 42/27
63-000 ŚRODA WLKP.

Tylko VIC 20
Wojciech Sławiński
02-758 Warszawa, ul. Burgaska 5 m 11


http://www.t2e.pl/reduksy

# WSZYSTKO DLA WSZYSTKICH

# refleks

NASZA OFERTA!!!



ASCOM TECHNOLOGIES (FAR EAST) PTE LTD

## PWPO-T „Refleks" Sp. z o.o. informuje,

że działa jako wyłączny przedstawiciel serwisowy na zasadzie zawartego kontraktu z ASCOM TECHNOLOGIES (FAR EAST) PTE LTD. Na zakupiony w tej firmie sprzęt wydawane jest w Polsce świadectwo jakości i udzielana jest roczna gwarancja, w czasie której funkcje gwaranta sprawuje na zasadzie wyłączności PWPOT „REFLEKS". Sprzęt zakupiony w ASCOM po odebraniu przesyłki przez użytkownika jest testowany i sprawdzany bezpłatnie w PWPOT „Refleks" Sp. z o.o.

**UŻYTKOWNIK OTRZYMUJE TYLKO DOBRY SPRZĘT!**

Ponadto „Refleks" udzieli Państwu wszelkich dodatkowych informacji zarówno handlowych, jak i technicznych (katalogi, cenniki itp.).
Kontakt: **Przedsiębiorstwo Wdrażania Postępu Organizacyjno-Technicznego „Refleks" Sp. z o.o. Dział Importu, 02-051 Warszawa, ul. Glogera 1 tel. (02) 659-20-41, (02) 659-39-22 tlx 817530 ref pl.**
Wysyłkowo z firmy ASCOM TECHNOLOGIES (FAR EAST) PTE LTD otrzymacie Państwo sprzęt mikrokomputerowy wysokiej jakości i w krótkich terminach dostawy:

**Oferta po atrakcyjnych cenach:**

— kompletne zestawy mikrokomputerów PC/XT 6/8/10 MHz, PC/AT 8/10/12 MHz, PC/38612/16/20 MHz oraz inne, jak np. mikrokomputery przenośne i najnowsze typy profesjonalnych mikrokomputerów,
— pełny asortyment kart CSKD, wyposażenia i akcesoriów umożliwiających samodzielne zbudowanie mikrokomputera lub rozszerzenie zestawu już posiadanego (karty główne, grafiki, kontrolery, karty obsługi wejść/wyjść, kable, obudowy, klawiatury, zasilacze),
— pełny asortyment urządzeń zewnętrznych, takich jak: monitory monochromatyczne i kolorowe (szeroka gama typów o różnej rozdzielczości), pamięci taśmowe, pamięci na miękkich dyskach i napędy dysków twardych (o bardzo dużej pojemności i krótkim czasie dostępu), różne typy drukarek firm: EPSON, CITIZEN, STAR, PANASONIC, Amstrad, różne typy ploterów i digitizerów,
**— nośniki magnetyczne,**
**— inne wyposażenie w środki techniki biurowej,**
**— urządzenia i przyrządy elektroniczne,**
**— urządzenia techniki video,**
**— elementy i podzespoły elektroniczne.**

**ASCOM TECHNOLOGIES/FAR EAST/PTE LTD**
**Republic of Singapore**
**45 Genting 05-02 Genting Warhouse Complex Singapore**
**1334 Republic of Singapore.**




**Przedsiębiorstwo Wdrażania Postępu Organizacyjno-Technicznego** **Sp. z o.o.**

# NOWOŚCI

# Polanglia — AMSTRAD

**NAJNIŻSZE CENY W EUROPIE**
**NA NAJLEPSZY SPRZĘT KOMPUTEROWY**

REWELACJA ROKU: NAJNOWSZY PC 640K PAMIĘCI W PEŁNI ZGODNY Z I.B.M.,
**AMSTRAD PC 1640 ECD**
ZGODNY Z EGA, HERCULES, MDA I CGA

**NOWA DRUKARKA** WYSOKIEJ KLASY ALE PO ZADZIWIAJĄCO NISKIEJ CENIE
**AMSTRAD DMP 3160** 160 CPS, 40 NLQ

**DRUKARKA ROKU 1987 :**
**AMSTRAD DMP 4000** 15", 200 CPS, 50 NLQ

NOWY PC/EDYTOR TEKSTU z LOCO SCRIPT 2, LOCO SPELL, (Slow. Ang.) LOCO MAIL
**AMSTRAD PCW 9512**

NOWY **SINCLAIR SPECTRUM** 128K PLUS 3
Z WBUDOWANĄ STACJĄ DYSKÓW

NAJTAŃSZY **SPECTRUM 128K PLUS 2** Z WBUDOWANYM MAGNETOFONEM

ORAZ NADAL N**AJPOPULARNIEJSZY PC W EUROPIE**
**AMSTRAD PC 1512** — PO ZNIŻONYCH CENACH
40 % RYNKU PC W WIELKIEJ BRYTANII - PODWÓJNIE NIŻ SAM I.B.M.

NA POWYŻSZY SPRZĘT JAK RÓWNIEŻ NA KOMPUTERY:
CPC 6128, PCW (JOYCE) I DRUKARKI **STAR** ZAKUPIONE U NAS
DOSTĘPNY JEST TAKŻE **SERWIS GWARANYJNY**

**AMSTRAD** SPRZEDAŁ NAJWIĘCEJ KOMPUTERÓW W EUROPIE (50 %)
**a POLANGLIA** W POLSCE

Zgodnie z warunkami aktualnej oferty firmy Polanglia, niniejszym zamawiam:

........................................................................ £ .........
........................................................................ £ .........
........................................................................ £ .........
PLUS kwota pobierana Przez Barclays Bank ........................................ = £ .....4.-

**RAZEM = £**

Załączam czek lub kopie zlecenia bankowego na przelew w/w sumy na konto Polanglia Ltd Nr. 70736805 w Barclays Bank PLC, Ealing Broadway Branch (kod 20-27-48), 53 The Broadway, LONDON W5 5JS, zrealizowanego w dniu ........ / ..... / ......

przez bank ........................................ oddział ........................................

Podpis wpłacającego .................................... Nazwisko i imię .................................... Data ....................................

NAZWISKO I IMIĘ ODBIORCY ........................................................................

PEŁNY ADRES ........................................................................

**ADRES: 171-175 Uxbridge Road, London W13 9AA**
**Tel: Londyn 840 1715 Telex: 946581 Fax: 840 7136**

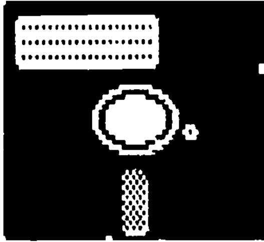
| | GIEŁDA BAJTKA (tys. zł) | PEWEX BALTONA (USD) | Sklep Bajtka (tys. zł) |
|---|---|---|---|
| **SINCLAIR** | | | |
| ZX81 | 25 | – | 39 |
| ZX Spectrum 48 kB | 75–80 | 115 | 140–180 |
| ZX Spectrum Plus | 110–120 | – | 180–250 |
| ZX Spectrum 128 kB + 2 | 240 | – | 299 |
| Drukarka SEIKOSHA GP50S | 65–80 | – | 99 |
| TIMEX 2048 | 95 | 146 | – |
| Joystick QUICKSHOT II | 4,5–7 | – | 5–9 |
| **COMMODORE** | | | |
| C-64 | 160 | 219 | 350 |
| C-128 | 240 | 299 | 480 |
| C-128D | 690 | – | 999 |
| Amiga z monit. kolorowym | 1,2 mln | – | 2500 |
| Magnetofon 1531 | 35 | 48 | 30 |
| Stacja dyskietek 1541 | 180 | – | 399 |
| Stacja dyskietek 1570 | 240 | 299 | 460 |
| Drukarka GP-500 | 150 | – | 149 |
| Dyskietki 5 1/4 (średnia jakość) | 0,7–1,5 | 3,5 | 0,3–1.5 |
| **ATARI** | | | |
| 65 XE | 110 | 125 | 170 |
| 130 XE | 165 | 199 | 340 |
| Stacja dyskietek 1050 | 180 | 187 | 350 |
| Drukarka 1029 | 170 | 199 | 299 |
| Atari 520 STM St. Dysk. 0.5Mb | 750–850 | – | 800 |
| **AMSTRAD** | | | |
| 464 z mon. monochromat. | 210 | – | 400 |
| 6128 z mon. monochromat. | 390 | – | 750 |
| 6128 z mon. kolor. | 450 | – | 950 |
| PCW 8256 | 600 | – | 1000 |
| Dyskietki 3” | 3,2–5 | – | 6–9 |
| Stacja dyskietek 3 do 464 | 320 | – | 500 |
| PC 1512 | 1,1 | – | 1150 |

## SUPER OKAZJA DLA INDYWIDUALISTY

Mikrokomputer 128 kB z 32-bitowym procesorem za jedyne 135 tys zł! Takiej pokusie mieli okazję ulec ci, którzy ostatniej soboty przyszli na warszawską giełdę.

W 1984 r. firma SINCLAIR zaprezentowała mikrokomputer nowego typu — QL, który zdecydowanie wyprzedzał swoje czasy i do dziś przewyższa możliwościami niejeden mikrokomputer domowy QL zawiera 2 mikroprocesory. 2 specjalizowane układy, możliwość adresowania do 1 MB bardzo dobry system operacyjny QDOS z możliwością sterowania wieloma „oknami” na ekranie i znakomitą wersję Basic'a. Posiada także 2 wbudowane microdriv'y o pojemności 100 kB każdy, z możliwością podłączenia 6 kolejnych. Niestety, okazało się, że ten system pamięci nie przyjął się i choć dostępne są prawie wszystkie języki programowania, nie ma na naszym rynku zbyt wielu programów użytkowych. Z tej przyczyny model QL służyć może wyłącznie poważnym pracom, a nigdy nie stanie się narzędziem w rękach zwolenników doskonalenia małpiej zręczności podczas zaliczania kolejnych programów rozrywkowych i gier. Może właśnie dlatego mimo niezaprzeczalnych zalet i niskiej ceny QL spokojnie czekał na giełdzie na nabywcę

Ciekawi jesteśmy skali trudności i mnogości sukcesów u wszystkich użytkowników tego mikrokomputera Proponujemy utworzyć klub posiadaczy QL. Piszcie do redakcji, będziemy pośredniczyć w wymianie doświadczeń i Programów.



# INDYWIDUALNY BANK DANYCH

Nazywam się **Sebastian Bienkiewicz**, jesiem uczniem i mam 13 lat. Posiadam mikrokomputer Texas Instruments 99/4A. Interesuję się chemią i innymi naukami przyrodniczymi. Posiadaczy takich samych mikrokomputerów proszę o informacje na temat Texas Instruments 99/4A. Mój adres: ul. Słowackiego 9a/1, 37-600 Lubaczów.

**Mirosław Moździerz**, technik budowlany, Mikrokomputer Atari 800 XL. magnetofon. Zainteresowania: radioelektronika, żeglarstwo. Poszukuje programów, wymieni literaturę. Adres: ul. Peresa 52/2, 53-420 Wrocław.

**Rafał Graczyk**, uczeń, 15 lat. Mikrokomputer Timex 2048, magnetofon, telewizor. Kilka gier i programów użytkowych. Zainteresowania: sport, filatelistyka. Adres: ul. Włocławska 17/27, Koło.

**Marcin Czerny**, uczeń, 13 lat. Mikrokomputer Commodore 16, magnetofon, drukarka, joystick. Gry, programy graficzne, programy muzyczne. Poszukuje sposobu na przekształcanie programów ze Spectrum na Commodore. Zainteresowania: majsterkowanie, kolarstwo. Adres: ul. Rybnicka 78/1, 44-100 Gliwice.

**Bartosz Choiński**, uczeń, 16 lat. Mikrokomputer Oric 1, Oprogramowanie: gry. programy edukacyjne — nauka Basica, nauka Logo, bank danych. Zainteresowania: komputery, motoryzacja. Adres: ul. Górna 26, Józefów.

**Ryszard Biliński**, student. 25 lat Mikrokomputer Amstrad CPC 464 printer-plotter, monitor monochromatyczny, joystick. Oprogramowanie: Fighter Pilot. Biorytm. Zainteresowania - chemia, astronomia, muzyka dawna. Wymiana programów, pomysłów i informacji. Adres: ui. Bema 27a/18. 81- 381 Gdynia.

**Michał Kozłowski,** uczeń, 12 lat. Mikrokomputer SVI 728, magnetofon. Zainteresowania: sport, elektronika Poszukuję oprogramowania do swojego nietypowego mikrokomputera. Adres: ul. Bończyka 21/8, 45-705 Opole.

**Szymon Miedziejko**. uczeń. 15 lat. Mikrokomputer Toshiba MSX magnetofon, joystick. Programy firmy Toshiba, Ocean, Kuma. Zainteresowania: fotografika, język angielski. Wymiana wiadomości o nowych, użytkowych sposobach wykorzystania komputerów. Adres: Oś. Chrobrego 9G/169, 60-681 Poznań.

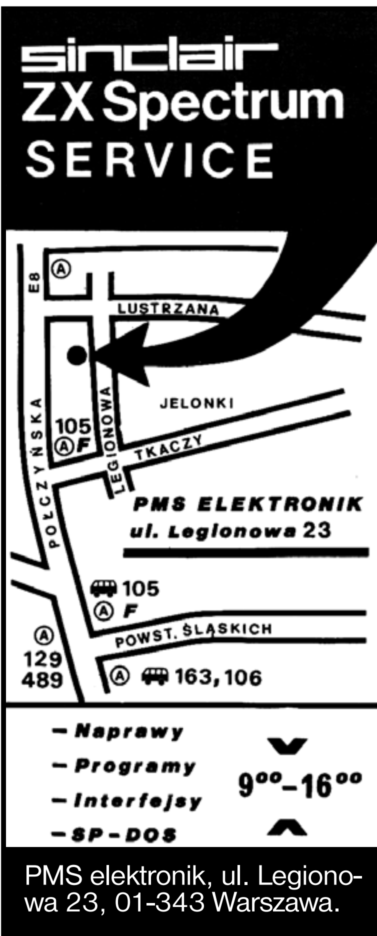
**Przemysław Ristok**. uczeń, 15 lat. Mikrokomputer ZX Spectrum+ , magnetofon. Programy zaczerpnięte z czasopism. Zainteresowania: elektronika, informatyka. Kontakt z innymi posiadaczami tego typu mikrokomputera. Adres: Bl. 31/15, 76-042 Rosnowo.

**Marek Jaworski**, inżynier, 36 lat. Mikrokomputer Spectravideo SV 328. Oprogramowanie: własne — w trakcie przygotowywania, firmowe — gry. Zainteresowania: wykorzystanie mikrokomputera w pracy zawodowej. Proponuję wymianę literatury na temat języka Basic, wersja 1,1 i SP/M w zastosowaniu do SV-328, tłumaczenia gier z języka angielskiego. Adres: ul. Mickiewicza 14/5, 66-400 Gorzów Wlkp.

**Piotr Wąs**, monter urządzeń elektrycznych, 19 lat. Mikrokomputer VZ 200. Oprogramowanie własne: zamiana liczb na różne systemy, firmowe-pokazowe w Basicu. Zainteresowania: informatyka w kontekście sposobów rozbudowy i programów użytkowych. Adres: ul. Wolności 534a/5, 41-800 Zabrze.

**Grzegorz Ikonomu**. Mikrokomputer VX VG 8020 MSX. Poszukuję oprogramowania, chciałbym nawiązać kontakt z „Fan Club MSX”. Adres: ul. Bartnicza 36/7 71-487 Szczecin.

**Marianna Maciąs**, prawnik, 34 lata. Mikrokomputer Atari 800 XL, monitor Neptun M 156B, magnetofon Atari 1010 joystick. Programy własne-graficzne i muzyczne, firmowe-gry, użytkowe: kopiujące, edukacyjne, narzędziowe. Zainteresowania: informatyka, literatura science-fiction. Wymiana oprogramowania i doświadczeń. Adres: skr poczt. 39 37-464 Stalowa Wola.




Programy Komputerowe,
Instrukcje I Udoskonalenia Techniczne
pocztą dla

• ATARI •
• AMSTRADA •
• COMMODORE'A •
• i IBM •

wysyła
AGENCJA MIKROKOMPUTEROWA
SOSNOWIEC P-157
tel. 699-649

# Drogi Bajtku!

***Ostatnio w prasie (P.T., Trybuna Ludu) ukazały się artykuły krytyczne pod adresem Pewexu i sprzedawanych tam Atari. Zdaniem autorów tych wypowiedzi ceny komputerów są zbyt wysokie w porównaniu z cenami w krajach zachodnich, a ponadto stwierdzono, że model 800XL jest przestarzały, natomiast „doskonały" 130XE „można nabyć np. w RFN w cenie 800XL", która obowiązuje w Pewexie.***

***Zastanawiam się nam tym porównaniem obu modeli, ponieważ z tego co wiem z waszego pożytecznego pisma, różnica tkwi jedynie w rozszerzonej do 128 KB pamięci 130XE, bo przecież podstawowe „kości" są takie same.***

***Proszę też o wyjaśnienie dotyczące przyszłości komputerów Atari w Polsce; czy fakt iż są one sprzedawane w Pewexie stanowi gwarancję dopływu oprogramowania i sprzętu towarzyszącego w dostatecznej ilości i czy w ogóle ten system ma przyszłość?***

***Jacek Altrych***
***ul. B. Polaka 7 m. 4***
***02-777 Warszawa***

Atari 130XE nie jest w porównaniu z modelem 800XL „rewelacją", lecz jedynie zmodernizowaną wersją o lepszej klawiaturze i większej pamięci. Oczywiście także cena tego modelu jest odpowiednio wyższa.

Ceny sprzedawanego przez siebie sprzętu ustala przedsiębiorstwo Pewex. Jest faktem, że Atarii 800XL jest komputerem obecnie już wyprzedawanyn na zachodzie. Reakcja Pewexu na związane z tym obniżki cen 800XL na rynku europejskim jest dość powolna i trzeba przyznać — ograniczona. Należy jednak zwrócić uwagę na ważną dla polskiego nabywcy cechę zakupu w Pewexie, jaką jest serwis gwarancyjny. Warto to uwzględnić przy kalkukowaniu cen. Nie zamierzam tu bronić polityki cenowej Pewexu; w interesie Bajtka jest bowiem, by komputery Atari kosztowały jak najmniej — co mogłoby tylko przysporzyć nam czytelników.

Porównanie cen w Pewexie z cenami na rynkach zagranicznych znaleźć można w rubryce „Giełda".

Pozycja 8-bitowych Atari jest w Polsce dość ugruntowana; niektóre ceny wykazują, że stają się one najpopularniejszymi komputerami domowymi w naszym kraju. Wpłynął na to niewątpliwie fakt możliwości zakupu tego sprzętu na miejscu, w kraju, po — było nie było — umiarkowanej cenie. Jeżeli chodzi o przyszłość rodziny XL/XE to sądzę, że nie należy się spodziewać nowych wersji rozwojowych w znaczący sposób odbiegających od znanych nam 130XE czy 65XE. Można natomiast liczyć na to, że będzie to sprzęt popularny, a przez to nie przysparzający zbędnych zmartwień swemu właścicielowi, przez jeszcze kilka lat. To samo można zresztą powiedzieć np. na temat komputerów C 64 i C 128.

Sam fakt sprzedaży Atari w Pewexie nie stanowi gwarancji dopływu oprogramowania. Trzeba podkreślić, że Pewex jest przedsiębiorstwem wielobranżowym, a nie wyspecjalizowanym w sprzedaży sprzętu elektronicznego. Dlatego sprzedawane są tylko podstawowe urządzenia peryferyjne. Jeżeli chodzi o programy, to Pewex oferuje raczej niezmienny zestaw kilkunastu do kilkudziesięciu pozycji; bez żadnych nowinek, bez olśniewających grafiką gier. Jak wykazuje doświadczenie, w dalszym ciągu podstawowym źródłem oprogramowania na naszym rynku pozostaje prywatny import i nielegalne (w świetle zagranicznych praw autorskich) kopiowanie. Wysoka pozycja Atari wskazuje jednak, że prywatny import będzie trwał. Jeżeli nie pojawią się i nie będą egzekwowane odpowiednie akty prawne, nabycie nielegalnych kopii również nie będzie problemem.

---

***Istnieje wiele tańszych drukarek od drukarki 1029 do Atari 800XL. Czy inną drukarkę, np. Seikosha GP 50S można w jakiś sposób podłączyć do tego komputera?***

***Tomek***
***(nazw. i adres do wiad. redakcji)***

Firma Atari opracowała własny standard drukarki, zarówno jeżeli chodzi o kształt łącza, jak o sposób transmisji danych. Z tego powodu producent proponuje używanie wyłącznie drukarek Atari.

Teoretycznie jest możliwe użycie dowolnej drukarki z dowolnym typem komputera, po wykonaniu odpowiedniego dla określonej pary urządzeń sprzęgu (interface'u) oraz użyciu właściwego programu transmisji danych. Swego czasu np. pokazywaliśmy w Bajtku jak „ożenić" w ten sposób ZX Spectrum z drukarką DZM-180 — co stanowi dość egzotyczne zestawienie.

Warto jednak zauważyć, że tanie drukarki (na przykład GP 50S) z reguły są mniej trwałe, mniej uniwersalne i produkują wydruki kiepskiej jakości.

---

***Jestem użytkownikiem Atari 800XL z magnetofonem firmowym, monitorem i joystickiem. Szczególnie interesuje mnie język Pascal. W związku z tym mam do Was kilka pytań:***

***Czy wersja KYAN PASCAL może być używana bez stacji dysków? Posiadam ów program na kasecie. Jeżeli nie, to czy powodem tego jest brak pamięci po wgraniu kompilatora do pamięci, czy też inne przeszkody?***

***Czy istnieje inna wersja Pascala, nie wymagająca stacji dysków?***

***Waldemar Kozaczuk***
***ul. Mickiewicza 48A m. 28***
***17-100 Bielsk Podlaski***

KYAN PASCAL wymaga zastosowania pamięci dyskowej. Używana jest ona do przechowywania tekstów źródłowych i kodu programów; poza tym ten typ pamięci zewnętrznej jest używany przez system wejścia/wyjścia w samym Pascalu. Dostępna wersja kasetowa została opracowana w Polsce na zlecenie jednej z prywatnych firm rozprowadzających oprogramowanie. Niestety nie jest ona nam znana i nie możemy wypowiadać się o jej jakości.

---

***Posiadam Atari 800XL; mam jedno pytanie. Mój program (liczący około 11KB) wykonuje się chyba 1000 godzin. Czy jest jakiś kompilator Basica dla Atari o takich możliwościach jak IS na Spectrum (200-krotne przyspieszenie)?***

***Zdzisław Muskała***
***Wola Wapowska***
***88-182 Bachorce***

Informacja o takim kompilatorze na Spectrum jest błędna. Najlepsze kompilatory na komputerach 8-bitowych pozwalają na uzyskanie kilkudziesięciokrot- nego przyspieszenia w stosunku do programów interpretowanych (powiedzmy ok. 40 razy). Większość (kiepskich z reguły) kompilatorów języka Basic pozwala na 1,5 — 2-krotne zwiększenie szybkości.

Jak widać, rozwiązanie Pańskiego problemu nie może polegać na użyciu wystarczająco dobrego kompilatora. Już wspominałem na łamach tej rubryki, że tego typu kłopotów prawie nigdy nie da się wyeliminować przez użycie kompilatora, szybszego komputera itp. Z reguły przyczyną zbyt długiego działania programu jest sama konstrukcja algorytmu. Nie znam rozwiązywanego przez Pana problemu, a jednak twierdzę, że program o objętości 11 KB zawsze daje się wykonać w rozsądnym czasie (o ile np. nie zawiera nieskończonej pętli). W razie potrzeby chętnie podyskutuję o szczegółach.

---

***Czy istnieją — i jakie — programy muzyczne dla komputera Spectrum?***

***Michał Dryński***
***Bytom***

Nawet jeżeli istnieją, to z pewnością są bardzo prymitywne; osobiście nie zetknąłem się z żadnym takim programem. Jest to oczywiste, ponieważ wbudowany monofoniczny generatorek de facto w ogóle nie pozwala na odtwarzanie muzyki. Wszelkie pakiety służące do tworzenia muzyki są przeznaczone z zasady dla komputerów wyposażonych w możliwość polifonicznej (wielokanałowej) generacji dźwięku, regulacji jego barwy i natężenia oraz niezależnego sterowania czasowego każdym kanałem — oto cztery cechy, których Spectrum nie posiada.

Pewne nadzieje w tej dziedzinie wiązać można z komputerem ZX Spectrum +2, który wyposażony jest w (też nie najdoskonalszy) trójkanałowy generator „zapożyczony" z komputerów Amstrad. Na razie jednak nie słychać o programach muzycznych na ten komputer.

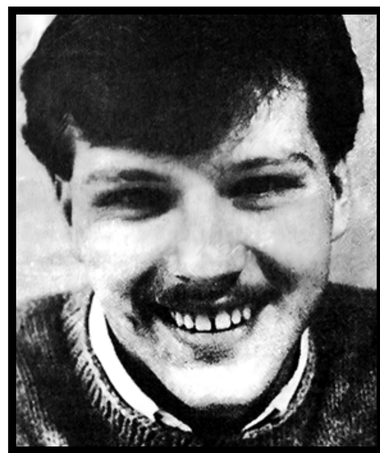
---

***Mam 16 lat, jestem uczniem I klasy Liceum Ogólnokształcącego i stałym czytelnikiem „Bajtka". W szkole posiadamy 5 mikrokomputerów Atari 800XL z magnetofonami. Na informatyce zajęcia mamy jednak tylko z jednym, gdyż do dyspozycji jest jeden telewizor. Mały dostęp do komputera sprawia, że nie mogę samodzielnie eksperymentować i wyszukiwać interesujących mnie rzeczy. Piszę dużo programów matematycznych i gier. Nie są to programy-giganty, ale spełniają przeznaczone im zadania. Czasami chodzę do kolegi, który jest posiadaczem Atari, ale to nie to samo, co posiadać je samemu.***

***Dużo uczę się analizując różne programy zamieszczone w czasopismach. Jednak są rzeczy, których nigdzie nie mogę wyczytać. Co zrobić, aby program po załadowaniu uruchamiał się automatycznie, tzn. bez użycia komendy RUN? Wiem, że na Spectrum jest to możliwe bez zagłębiania się w Asembler.***

***Czy można programować klawisz RESET, tak jak dzieje się to z klawiszami HELP, OPTION czy START? Jeżeli tak, to w jaki sposób?***

***Arkadiusz Sochan***
***Dąbrowa Górnicza***



**Na listy czytelników odpowiada Marcin Waligórski**

Pierwszą część tego listu zamieszczam celowo, gdyż jest niezwykle pouczająca. Oto kolejny przykład zasady „najpierw wydaję pieniądze, potem będę myślał". Nie wiem, czy wymieniona szkoła zakupiła komputery z własnych funduszy, czy też otrzymała je w darze; wiem jednak na pewno, że w decyzji tej zabrakło odrobiny zdrowego rozsądku. Czy trudno jest przewidzieć, że komputer bez monitora nie będzie mógł być używany? Czy nie można było — za te same pieniądze — zakupić czterech komputerów i czterech monitorów? Przez bezmyślność cierpią przede wszystkim uczniowie — którym być może odtąd słowo „komputer" kojarzyć się będzie z kosztownym urządzeniem, zamkniętym na klucz w szafie, którego nie można nawet uruchomić, a co dopiero używać w nauce...

List ten dedykuję wszystkim, którzy w najbliższym czasie chcieliby wyposażać placówki oświatowe w sprzęt komputerowy.

Odpowiedzi na pytania: Niestety, rozwiązanie problemu automatycznego uruchamiania programu nie jest tak łatwe, jak w przypadku Spectrum. Programy firmowe są wyposażone w specjalny „loader", który steruje ładowaniem programu, by następnie go uruchomić. Inną metodą może być wczytanie programu w obszar tzw. wektorów systemu operacyjnego i zmiana tych ostatnich do własnego celu. Obie te metody wymagają pewnego zaawansowania i znajomości kodu maszynowego (lub asemblera).

Działanie klawisza RESET nie może być zmienione programowo. W odróżnieniu od innych klawiszy funkcyjnych, działanie to jest bowiem wywoływane sprzętowo — RESET jest po prostu przyciskiem zerującym procesor.

---

***Czy i kiedy będzie wydana książka o Atari 800XL, w takiej formie, jak „Przewodnik po ZX Spectrum".***

***Adam Michalski***
***ul. Kołłątaja 1577***
***45-064 Opole***

Jak na razie brak tego typu wydawnictw. W chwili obecnej na rynku pojawia się literatura na temat komputera Spectrum, który był pierwszym popularnym komputerem domowym w Polsce. Biorąc pod uwagę powszechnie znane kłopoty poligrafii, nie należy się spodziewać szybkiego wydania pozycji książkowych traktujących o Atari. Niestety! Pozostaje mieć nadzieję na jedną lub dwie „jaskółki" w mniejszym formacie i objętości, za to o krótszym cyklu wydawniczym. Kto pierwszy?

*Marcin Waligórski*


http://www.t2e.pl/reduksy

W dniach 13 i 14 listopada 1987 roku odbędzie się pierwsze w Polsce Międzynarodowe Spotkanie Klubów Mikrokomputerowych. Impreza organizowana przez Stowarzyszenie Mikrokomputerowe ABA- KUS i Centrum Videoinformatycz- 3 ne Warszawskiego Ośrodka Kultury, ma na celu umożliwienie kontaktów i wymiany doświadczeń animatorom Klubów Komputerowych z kraju i zagranicy.

Program imprezy obejmuje:

1. Wystawę klubów, na której zaprezentuje swoje osiągnięcia, na indywidualnych stanowiskach, dziesięć najlepszych klubów z kraju i zagranicy.
2. Sejmik przedstawicieli Klubów Mikrokomputerowych, który odbędzie się w drugim dniu imprezy i będzie miał na celu przedyskutowanie problemów związanych z działalnością Klubów i sposobów ich rozwiązania.
3. Konkurs z cennymi nagrodami na wykonanie zadanych programów komputerowych w dwóch kategoriach:
   I dla zaawansowanych na komputerach 16-bitowych kompatybilnych z IBM PC
   II dla początkujących na komputerach 8-bitowych: Atari, Commodore, Amstrad, Spectrum.
4. Seminaria z udziałem przedstawicieli Klubów Mikrokomputerowych z następujących tematów:
   1) formy działalności klubów,
   2) rola klubów w rozwoju mikroinformatyki,
   3) wybrane problemy prawne z działalności klubów.

Przewidziano również rozdanie honorowych wyróżnień „ABAKÓW" 87 Stowarzyszenia Mikrokomputerowego ABAKUS za rok 1987, w następujących kategoriach:

I Najlepszy Klub Mikrokomputerowy w Polsce
II Najciekawszy program komputerowy
III Najlepsza publikacja o tematyce mikroinformatycznej

W czasie trwania imprezy czynne będzie stoisko z prasą i książkami. Miejscem spotkania będzie Warszawski Ośrodek Kultury ul. Elektoralna 12, gdzie również mieści się Biuro Organizacyjne tel. 20-48-38 w. 205 pokój nr 3. Wszystkie kluby i osoby zainteresowane udziałem w imprezie proszone są o zgłoszenie z podaniem posiadanego sprzętu, oprogramowania, filmów video (o tematyce mikroinformatycznej), materiałów szkoleniowych, literatury, krótkiego zarysu historii i działalności klubu, do Biura Organizacyjnego „I Międzynarodowego Spotkania Klubów Mikrokomputerowych".

**ABAKUS**

# NIE ZAPOMNIJ O PRENUMERACIE!

**Warunki prenumeraty „Bajtka"**

Terminy dokonania wpłat:
— do dnia 10 listopada: na styczeń, I kwartał, I półrocze roku następnego oraz cały rok następny,
— do dnia 1-go tego miesiąca, który poprzedza okres prenumeraty (roku bieżącego).

Ceny:
— kwartał: 300 zł, półrocze: 600 zł, rok: 1200 zł.

Prenumerata dla osób prywatnych:
— czytelnicy mieszkający w miastach, gdzie istnieją Oddziały RSW „Prasa-Książka-Ruch" opłacają prenumeratę w urzędach pocztowych nadawczo-oddawczych (właściwych dla miejsca zamieszkania prenumeratora) wpłacając na rachunek bankowy miejscowego Oddziału RSW „Prasa-Książka-Ruch"
— czytelnicy mieszkający na wsi i miejscowościach gdzie nie ma Oddziałów RSW „Prasa-Książka-Ruch" opłacają prenumeratę w urzędach pocztowych i u doręczycieli.

Prenumerata dla instytucji i zakładów pracy.
— zamówienia na prenumeratę przyjmowane są bezpośrednio przez Oddziały RSW „Prasa-Książka-Ruch" — a w miejscowościach, gdzie nie ma Oddziałów RSW prenumeraty można dokonać w urzędach pocztowych.

Prenumerata „Bajtka" zagranicę:
— prenumeratę ze zleceniem wysyłki przyjmuje RSW ..Prasa-Książka-Ruch" Centrala Kolportażu Prasy i Wydawnictw, ul. Towarowa 28. 00-958 Warszawa, konto NBP XV Oddział w Warszawie Nr 1153-201045- -139-11. Prenumerata ta. (z wysyłką pocztą zwykłą) jest droższa od prenumeraty krajowej o 50 % dla osób prywatnych i o 100% dla instytucji i zakładów pracy.

Studio komputerowe
**ATARI-BAJT**
**ATARI ● AMSTRAD**
**COMMODORE ● SPECTRUM**
oferuje
programy użytkowe,
edukacyjne, gry, opisy,
interfejsy do ATARI
W-wa tel. 20-80-34
lub 17-69-48
katalogi gratis

# COŚ DLA AMBITNYCH
## —CZYLI KĄCIK FIZYKI KOMPUTEROWEJ

Profesjonalne czasopismo „Fizyka w szkole" dwumiesięcznik Ministerstwa Oświaty i Wychowania zdecydowało się otworzyć na swoich łamach kącik fizyki komputerowej. Trzeba powiedzieć, że inicjatywa ta powstała dzięki aktywności nauczycieli. W ostatnim bowiem czasie do redakcji „Fizyki w szkole" napłynęło wiele programów, wykonanych prze nauczycieli i ich uczniów, dotyczących różnorodnych zagadnień z fizyki. Szereg z tych zagadnień nie da się rozwiązać na poziomie szkoły średniej za pomocą tradycyjnej matematyki, wymaga bowiem zastosowania metod (czasem nawet zaawansowanych jak w przypadku nieliniowych równań ruchu) wyższej matematyki. Komputer otwiera przed nami nowe możliwości — odciążając od nużących obliczeń, pozwala nareszcie fizyce „wyjrzeć zza węgła". To co było niedostępne, teraz za sprawą komputera można nawet obejrzeć na ekranie monitora.

Oczywiście coś za coś; takie podejście wymaga od nas umiejętności skutecznego programowania. Często bywa to jednak łatwiejsze, a nawet przyjemniejsze niż posługiwanie się tradycyjnym podejściem matematycznym. Nie chcę broń Boże przez to powiedzieć, że matematykę należy rzucić w kąt; chcę tylko podkreślić, iż otwierają się przed nami dodatkowe możliwości pozwalające zrozumieć nawet złożone problemy z fizyki już na wczesnym etapie zaawansowania matematycznego. Na przykład równanie opisujące ruch zwykłego wahadła matematycznego wprowadza nas w obszar całek eliptycznych, (o ile wychyleń wahadła nie można potraktować jako małe). Któż z nas odważyłby się zreferować to uczniom szkoły średniej? Tymczasem podejście numeryczne z zastosowaniem komputera pozwala bez większego trudu, krok po kroku z dobrym przybliżeniem rozwiązać równanie ruchu takiego (nieliniowego) wahadła matematycznego, a rozwiązanie (dla różnych wartości charakterystycznych parametrów) przedstawić na ekranie monitora.

Wielu nauczycieli uchwyciło tę unikalną możliwość i ze sporą już wprawą używa metod numerycznych do rozwiązywania ciekawych problemów fizyki. Kącik fizyki komputerowej pragnie udostępnić swoje szpalty nie tylko nauczycielom i uczniom, ale także wszystkim tym. którzy zechcą swoje programy z fizyki przedstawić szerszemu ogółowi. Redagowanie kącika powierzono niżej podpisanemu wierząc, iż będzie służył swoim doświadczeniem wszystkim, którzy zechcą współpracować z kącikiem. Muszę przyznać, iż z wiarą przystępuję do pracy, którą uważam za użyteczną ze wszech miar. Chciałbym, aby fizyki nie traktowano jako wiedzy tajemnej. Fizyka jest dla ludzi, a zwłaszcza dla ludzi młodych, pełnych fantazji. Chciałbym pokazać że fizyka, która się przecież dzieje wokół nas i w nas, może dać się lubić; drogę do tego celu widzę m.in. właśnie w metodach komputerowych. A zatem do zobaczenia w kąciku fizyki komputerowej.

**Ryszard Kutner**
**Wydział Fizyki U.W.**
**Instytut Kształcenia Nauczycieli**
**w W-wie**

**Zgodnie z wcześniejszymi zapowiedziami**

**P.Z. GALLECH**

**z siedzibą w Miechowie serdecznie zaprasza wszystkich zainteresowanych do swojego salonu wystawowego otwartego w każdy dzień roboczy.**

**Specjaliści naszej firmy prezentują:**
**— komputery 32-bitowe kompatybilne z IBM PC/AT,**
**— wielodostęp pod systemem operacyjnym XENIX,**
**— języki bez danych pod systemem operacyjnym XENIX/ INFORMIX SQL, FOXBASE+ — stuprocentowa zgodność ze standardem DBASE III plus,**
**— oprogramowanie baz danych pracujących w sieciach (SQL BASE, DBASE III plus, CLIPPER AUTUMN 86)**
**— kompilatory i interpretery języków (C, MS—FORTRAN, MS-PASCAL, MS-BASIC),**
**— procedor tekstu (LYRIX),**
**— sieci D-LAN i E-LAN (typu D-LINK i ETHERNET),**
**— sieciowe systemy operacyjne (IBM PC LAN PROGRAM, D-LINK NETBIOS EMULATOR, D-LINK NETWARE DRIVER, ADVANCED NETWARE 286)**

**SALON WYSTAWOWY MIEŚCI SIĘ W BUDYNKU FIRMY W MIECHOWIE PRZY UL. RACŁAWICKIEJ 31.**

**SERDECZNIE ZAPRASZAMY**



# NEMONIK — ELWRO

## Wrocław

Klub wkroczył bardzo pewnie w drugi rok działalności. Powstał w czerwcu 86 r. z inicjatywy ZZ ZSMP i KZ PZPR w Zakładach Elektronicznych "Elwro" we Wrocławiu. Program działania przedstawiło i błyskawicznie (nie ma się co dziwić — wiado mo: firma zobowiązuje) wcieliło w życie kilku zapaleńców, pracowników "Elwro" pod wodzą inz. WIESŁAWA SALIKA.

W ubiegłym roku zajęcia odbywały się w kilku grupach. Królowały programy w języku PASCAL i LOGO. Dwie grupy dzieci pracowników "Elwro" bawiły się w LOGO i BASIC. W większości byli to uczniowie ostatnich klas szkół podstawowych. Hasłem wywoławczym było zachęcenie młodzieży poprzez komputer do nauki fizyki i matematyki Dla wielu była to okazja powtórzenia sobie materiału przed egzaminami wstępnymi do szkół średnich.

Okazało się, że matematyka i fizyka wspomagane przez komputer są jeszcze prostsze. Stałymi bywalcami (czyli takimi, których od komputera trzeba kijem gonić — jak mawiają prowadzący te zajęcia) byli min. **Wojtek Żukrowski** — dziś już uczeń wrocławskiego liceum, **Rafał Zarzycki** i **Paweł Wołek**. Pierwsi dwaj okazali się absolutnymi fanatykami grafiki komputerowej. Wojtek stworzył m.in. kapitalny — zdaniem fachowców — program w obracanie brył w przestrzeni. Z kolei Paweł po długim, cierpliwym, "ślęczeniu" przy komputerze napisał "cwany" program w LOGO na kreślenie dowolnej funkcji ciągłej.

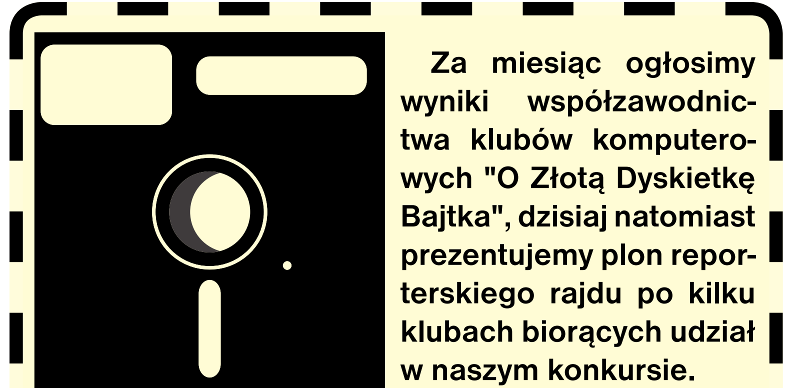
Na początku września br. obok do tej pory pracujących grup stworzyła się nowa, skupiająca dzieci w wieku 10—13 lat. Będą one poznawać tajniki programowania w LOGO. Zainteresowanie było wielkie. Rodzice dzwonili i dopytywali się kiedy wreszcie będą zapisy, przyprowadzali swoje pociechy na miejsce, by udowodnić, jaki mają one pociąg do komputera. Na razie nie ruszyła jeszcze żadna grupa dorosłych W związku z rozpoczęciem na dużą skalę przez "Elwro" produkcji komputera "Junior", w tej chwili każdy tu zatrudniony ma dostęp do komputerów na hali. Ćwiczenia mogą więc się odbywać nie jako "przy okazji" normalnej pracy. Chociaż wszyscy przyznają, że to właśnie "MNEMONIK" dal wielu pierwszego wielkiego "kopa" by zaiąć się komputerem nie tylko dla zabawy, ale i dla wykorzystania go w życiu codziennym i pracy.

Jednym z organizatorów "MNEMONIKA" jest **Zdzisław Kolan**, zastępca dyrektora Elektronicznych Zakładów Naukowych — a więc średniej szkoły elektronicznej sąsiadującej przez płot z "Elwro" i mającej z tą firmą wielorakie bardzo ścisłe związki. Siedziba klubu znajduje się właśnie na terenie EZN. W pracowniach jest 71 — komputerów — głównie Spectrum, choć nie brak tu też i innych "bardziej poważnych" maszyn. Niewykluczone, że w najbliższym czasie w ramach klubu ruszy tu nauka języka obcego przy pomocy komputera. W tej chwili szuka się odważnego filologa.

Wręcz niezwykłym wydarzeniem w życiu klubu była sesja, jaka odbyła się w czerwcu br. w gmachu NOT. Duszą i mózgiem przedsięwzięcia był inż. Salik, a tematem "Zastosowanie komputerów w nauczaniu dzieci niepełnosprawnych". Przyszli tłumnie dyrektorzy zakładów opiekuńczych i wychowawcy. Materiały jakie zostały przedstawione były zdumiewające. Okazuje się. ze juz po roku doświadczeń obserwować można u chorych znaczące postępy.

*Wojciech Głuch*

**"MNEMONIK", 53-238 Wrocław ul. Ostrowskiego 30**






# FENIKS

## Recz

Jak to często bywa, o tym że młode małżeństwo Walaków osiedliło się w niewielkim a uroczym Reczu, zadecydowało mieszkanie.

Zaoferowane przez naczelnika samodzielne lokum skłoniło parę absolwentów Uniwersytetu Łódzkiego do podjęcia pracy w miejscowej szkole podstawowej. Inna sprawa, że mgra **Mariusza Walaka,** specjalistę z zakresu zastosowań matematyki, nie usatysfakcjonowało powierzenie mu prowadzenia... szkolnej świetlicy. Chętnie przeto przystał na propozycję miejscowej władzy dotyczącą objęcia stanowiska dyrektora Miejsko--Gminnego Ośrodka Kultury. „**Wówczas** — jak nam mówi — **takiej nieco większej, zaniedbanej i wiejącej nudą świetlicy**".

Tak było przed kilku laty. Dziś MGOK w Reczu jest placówką, której nie powstydziłby się wojewódzki Gorzów.

Najchętniej opowiada jednak o działającym tu dopiero nieco ponad rok klubie mikrokomputerowym „FENIKS". Ze sposobu prezentowania niewielkich jeszcze przecież osiągnięć jego członków widać, że mgr Walak jest wielkim pasjonatem informatyki i komputerów. Ową pasją „zaraził się" jeszcze podczas studiów, ściślej — w trakcie rocznej praktyki z zakresu programowania i obsługi komputerów, którą odbywał w Stanach Zjednoczonych, w Control Date Institute.

W Reczu początkowo nie ujawniał się je swoimi zainteresowaniami. Dopiero po wykazaniu się efektami uzyskanymi w tradycyjnych dziedzinach działalności ośrodka zaproponował rozszerzenie jej o popularyzację, szczególnie wśród młodzieży szkolnej, informatyki i mikrokomputerów.

Pierwszy mikrokomputer UNIPOLBRIT z monitorem NEPTUN 557 i dwoma manipulatorami, zakupił dla ośrodka w marcu ubiegłego roku. Gdy w szkole podstawowej ogłosił zapisy do nowo powstającej sekcji zainteresowanie uczniów przekroczyło wszelkie oczekiwania. „**W ciągu godziny** — opowiada — **zgłosiła się ponad setka. Mogłem przyjąć nie więcej niż czwartą część. Ostateczną listę wybrańców ustaliłem w oparciu o wyniki ułożonego przeze mnie testu**".

Największy kłopot wynikał stąd, że był jedyną osobą w Reczu posiadającą przygotowanie fachowe w zakresie informatyki i obsługi komputerów. Mając wiele innych obowiązków zawodowych, na zajęcia w grupach mógł poświęcić w tygodniu nie więcej niż trzy popołudnia.

Pierwsze miesiące funkcjonowania klubu „FENIKS" (taką nazwę przyjęli od 1 maja ub. roku) ukazały narastające zainteresowanie jego działalnością wśród miejscowej młodzieży. Członkowie klubu już w czerwcu zaczęli próbować swych sił w rysowaniu schematów blokowych.

Dostrzeżono to także w stolicy województwa. Miejsko-Gminny Ośrodek Kultury w Reczu otrzymał dalsze dotacje. Z początkiem nowego roku szkolnego klub wzbogacił się o kolejne mikrokomputery: ZX Spectrum, ZX Spectrum Plus oraz ZX Spektrum 128. Zakupiono także monitory i całe należne oprzyrządowanie i wyposażenie, z kasetami gier włącznie.

Na początku bieżącego roku stan posiadania „FENIKSA" powiększył się o mikrokomputer Atari 800 XL ze stacją dysków Atari 1050 i magnetofonem Atari 1010. Pozwoliło to dyr. Walakowi na ogłoszenie dodatkowych zapisów.

Od kilku miesięcy do klubu należy blisko setka uczennic i uczniów ze szkoły podstawowej oraz szkół średnich. Skupieni są w trzech grupach gier komputerowych, grupie programowania oraz dwóch grupach matematycznych. Mariusz Walak ma już pomocników Zajęcia z gier prowadzi przeszkolony wstępnie przez niego młody pracownik miejscowej fabryki sprzętu okrętowego, zaś grupy matematyczne — emerytowany magister matematyki

— **Nie obraziłbym się** — powiada mgr Mariusz Walak — g**dyby w Reczu i innych miejscowościach naszego województwa powstały „konkurencyjne" kluby. Cieszę się, że w pobliskim Choszcznie powstaje klub „Atari". Oby takich inicjatyw było jak najwięcej. U progu XXI wieku nie do pomyślenia być powinno, by absolwent szkoły podstawowej nie siedział nigdy przed komputerem.**

*Zygmunt Kowalski*

**„Feniks", 73-210 Recz, ul. Ratuszowa, Miejsko-Gminny Ośrodek Kultury.**



# SYNTAX ERROR

## Koszalin

Uważamy, że słowo „klub" w naszym przypadku brzmi zbyt dumnie — jesteśmy grupą ludzi, która chce być dobrze przygotowana do zawodu i nie bać się kiedyś komputera — mówi jeden z załóżcieli, ANDRZEJ MATWIEJEWICZ, ale skoro jest w nazwie „klub" niech tak już zostanie.

Jest ich 12. W większości są studentami czwartego roku Budownictwa Lądowego Wyższej Szkoły Inżynierskiej w Koszalinie. Swój klub „SYNTAX ERROR" założyli rok temu. Pochodzą głównie z miasteczek i wsi z okolic Koszalina i Słupska. Wszyscy należą do ZMW, ale to nie znaczy, że są grupą zamkniętą. Łączy ich zafascynowanie najnowszą techniką.

Program ich studiów obejmuje między innymi elektroniczne techniki obliczeniowe. W ciągu dwóch semestrów poznali tajniki programowania w FORTRAN-ie i BASIC-u pracując na czeskim komputerze SM-40.

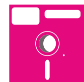
— Ponieważ SM-40 jest już przestarzałym komputerem i wychodzi z użycia, uznaliśmy, że jest to za mało. W przyszłości będziemy przecież konstruktorami, przyjdzie nam pracować na komputerach o dużej pojemności, takich np. jak IBM, Amstrad — mówi Andrzej. — Nie to, że my chcemy poznawać technikę komputerową, my musimy ją poznawać. Inna sprawa, że nie wszyscy byli tym zainteresowani...

Wybrali nawet prezesa, **Andrzeja Paruszewskiego**. Musieli jednak odpowiedzieć sobie na pytanie, skąd wziąć komputery. Mogli zwrócić się o pomoc do uczelni, która dysponuje różnym sprzętem (ma nawet swój klub mikrokomputerów, tyle że jak dotąd — bardziej adresowany do kadry). Są przekonani, że uzyskaliby taką pomoc, ale wówczas byłby dostępny na terenie uczelni, w wyznaczonych godzinach itp. Dlatego wybrali drugie wyjście. Poszukują na własną rękę komputerów, które będą stały w akademiku do wykorzystania praktycznie — o każdej porze.

Skończyło się na tym, że pracowali na ZX-Spectrum, należącym do jednego z nich, **Mirosława Syczą**. W tym roku jednak każdy z nich zarobił nieco złotówek na wakacjach, właściciel ZX-Spectrum sprzeda swego „staruszka", kupią wspólnie ATARI-800 XL. Spółka „SYNTAX ERROR" nie zamierza zatem składać broni i liczy po cichu, że uda się jej znaleźć na uczelni jakieś pomieszczenie dla siebie.

*Waldemar Ćwieka*

**„SYNTAX ERROR", 75—507 Koszalin, ul. Rejtana 15..**



# ZŁOTY AMSTRAD

## Złoty Stok

Mimo pozornego oddalenia Złotego Stoku (leży na obrzeżach Kotliny Kłodzkiej, nad granicą polsko-czechosłowacką) od „wielkiego świata", miasteczko wcale nie zamierza dać się sprowadzić do roli zapomnianego kopciuszka. Jest to możliwe dzięki istnieniu tutaj zakładów tworzyw i farb, których nigdy nie trzeba było usilnie namawiać, by włączały się do codziennego życia mieszkańców Złotego Stoku. Jednym z tego przejawów było zorganizowanie i wyposażenie w Zakładowym Domu Kultury klubu komputerowego.

Zakłady zakupiły 5 komputerów „Amstrad" (w tym dwa z monitorami kolorowymi) stacje dysków 3, drukarkę D100. Pracownia komputerowa jest czynna od wczesnych godzin popołudniowych i wtedy mają zajęcia uczniowie złotostockich szkół) do poźnej niejednokrotnie nocy Przychodzi tu młodzież i dorośli nie tylko ze Złotego Stoku, ale i okolicy. W czasie wakacji z komputerów korzystały dzieci przebywające tu na koloniach i obozach.

Klub komputerowy „ZŁOTY AMSTRAD" ma na swoje zajęcia wtorki i czwartki. Prowadzi go młody pracownik działu informatyki w Zakładach Tworzyw i Farb — **Krzysztof Woliński** W tej chwili realizowana jest nauka programowania w LOGO. Potem członkowie klubu — będą poznawać inne języki — przede wszystkich RASCAL-a. W ramach zajęć klubowych nie prowadzi się gier komputerowych. Z „salonu" gier można korzystać w innym czasie.

W związku z tym, ze klub znajduje się w conajmniej] 100 kilometrowym oddaleniu od najbliższego innego klubu komputerowego rozwija bardzo ożywioną korespondencję z klubami w kraju i za granicą. Ze zrozumiałych względów najbliższi mu są ci, którzy pracują na Amstradach — udało się nawiązać kontakty z takimi ośrodkami we Francji i Anglii. Czekają na odpowiedź z USA. Jak się przekonali, droga korespondencyjna jest całkiem wygodna i pewna. Zamierzają tę formę kontaktów i wymiany doświadczeń rozszerzać.

Czują się pionierami informatyki, to zachęca do wytrwałej pracy i pozwala pozbyć się „kompleksu prowincji" „Złoty Stok" który ma „ZŁOTEGO AMSTRADA" wcale nie musi być gorszy od Warszawy czy Wrocławia.

*Wojciech Głuch*

**ZŁOTY AMSTRAD, 57-250 Złoty Stok, ul. 1-go Maja 10**

# KUBUŚ TAŃCZY BREAK DANCE

*Cześć Maluchy!*

*Pamiętacie tańczącego Kubusia Literkę z pierwszego numeru "Bajtka" z 1986 roku? Dzisiaj wrócimy do tego pomysłu. Tym razem jednak nasz przyjaciel Kubuś będzie wykonywał znacznie bardziej skomplikowane ewolucje.*

Zapewne nie wszyscy wiecie, że Kubuś ma brata bliźniaka — Barnabę, który jest znakomitym tancerzem. Kubuś także postanowił nauczyć się tańczyć. Poprosił więc brata o udzielenie mu kilku lekcji.

Na ekranie zobaczymy więc Barnabę i Kubusia. Gdy Barnaba wykona jakiś ruch, Kubuś będzie go musiał — najszybciej jak tylko może — powtórzyć. Oczywiście z Waszą pomocą.

Wiemy jak ma wyglądać nasz program. W takim razie możemy już zabrać się do pracy. Na początek wymyślimy sobie jak mają wyglądać Kubuś i Barnaba w pozycji, że tak powiem, zasadniczej.

```
1000 PRINT "          %%   "
1010 PRINT "         (..)  "
1020 PRINT "         (--)  "
1030 PRINT "          )(   "
1040 PRINT "         /XX\  "
1050 PRINT "         IXXI  "
1060 PRINT "         IXXI  "
1070 PRINT "          II   "
1080 PRINT "          II   "
1090 PRINT "         <II>  "
```

Teraz spróbujmy wymyślić taneczne pozy sympatycznych braci. Oczywiście nie musimy już rysować całych postaci, a jedynie ich fragmenty.

```
2000 PRINT "         (\/)  "
2010 GOTO 1030
3000 PRINT "       ==XX\   "
3010 PRINT "          XXI  "
3020 PRINT "          XXI  "
3030 GOTO 1070
4000 PRINT "          /XX== "
4010 PRINT "          IXX  "
4020 PRINT "          IXX  "
4030 GOTO 1070
5000 PRINT "       A IXXI  "
5010 PRINT "       ---XXI  "
5020 PRINT "           I   "
5030 PRINT "           I   "
5040 PRINT "           I>  "
5050 GOTO 1100
6000 PRINT "         IXXI A "
6010 PRINT "         IXX--- "
6020 PRINT "         I      "
6030 PRINT "         I      "
6040 PRINT "        <I      "
6050 GOTO 1100
```

W linii 2000 umieszczony jest uśmiech tancerza, w liniach 3000-3020 podniesiona prawa ręka itd. Za każdym razem, dzięki instrukcji skoku, program powraca do rysowania odpowiedniego fragmentu postaci umieszczonej w liniach 1000-1090. Po narysowaniu nóg, program powraca do linii o numerze 1100, w której piszemy:

```
1100 RETURN
```

w ten sposób tworzymy podprogram, do którego będziemy się mogli odwołać instrukcją **GOSUB**. Trzeba jeszcze powiązać program rysujący pierwszą postać z fragmentami obrazującymi ruch. Wybór odpowiedniej pozycji będzie zależał od wartości zmiennej tekstowej **ruch$**

```
1015 IF ruch$="q" THEN GOTO 2000
1035 IF ruch$="a" THEN GOTO 3000
1036 IF ruch$="s" THEN GOTO 4000
1045 IF ruch$="z" THEN GOTO 5000
1046 IF ruch$="x" THEN GOTO 6000
```

Jeśli teraz w pętli umieścimy instrukcję odczytującą stan klawiatury — **INKEY$** oraz skok do podprogramu rysowania Kubusia, nasz bohater zacznie się nareszcie poruszać zgodnie z naszymi poleceniami.

```
20 LET ruch$=""
39 REM ***** stopien trudnosci *****
40 PRINT "Wybierz stopien trudnosci (1-5)."
200 FOR k=1 TO stopien
210   LOCATE 1,1
510   LET ruch$=INKEY$
515   LET punkty=punkty-1
520   IF ruch$="" THEN GOTO 510
525   IF ruch$<>kruch$ THEN GOTO 7000
530   GOSUB 1000
540 NEXT k
```

Mamy więc już Kubusia, a gdzie Barnaba? Jego ruchami musi sterować — w sposób losowy — komputer. Program rysujący Barnabę umieścimy oczywiście wewnątrz tej samej pętli. Komputer losuje najpierw liczbę z zakresu 1 - 5, a następnie przy pomocy instrukcji **RESTORE** i **READ** oraz danych w linii **DATA** (umieszczonych już poza pętlą) dobiera odpowiadająca jej wartość zmiennej **ruch$** Pojawia się tu jeszcze jedna zmienna: **kruch$**, oznaczająca ruch komputera, będzie ona potrzebna do porównania ruchów Barnaby i Kubusia.

```
220 LET los=INT(RND(1)*5)+1
230   RESTORE
240    FOR i=1 TO los
250      READ kruch$
260    NEXT i
265   IF kruch$=ruch$ THEN GOTO 220
270   LET ruch$=kruch$
280   GOSUB 1000
```

Tym razem na ekranie tańczą już obydwaj bracia, pierwszy wykonuje przypadkowe ruchy, drugim zaś możemy kierować naciskając odpowiednie klawisze. Zauważmy jednak, że Kubuś (postać w dolnej części ekranu) pojawia się dopiero wówczas, gdy naciśnięty zostanie po raz pierwszy jakiś klawisz. Wynika to oczywiście z warunku zawartego w linii 520. Uniknąć tej niedogodności możemy dopisując przed pętlą:

```
150 LOCATE 1,1
160 GOSUB 1000
170 PRINT:PRINT
180 GOSUB 1000
```

Spróbujmy teraz uruchomić ten program. Po chwili zabawy stwierdzimy, że Barnaba nie za każdym razem zmienia pozycję. Dlaczego? Po prostu może się zdarzyć, że komputer kolejno wylosuje te same liczby i w ten sposób pozornie pozostanie bez ruchu. Wykluczymy taką możliwość dopisując linię:

```
265   IF kruch$=ruch$ THEN GOTO 220
```

Na razie Kubuś i Barnaba poruszają się niezależnie od siebie, a przecież na początku stwierdziliśmy, że Kubuś ma powtarzać ruchy Barnaby. Tak więc za pomyłkę powinna naszego przyjaciela spotkać kara. Teraz przyda nam się zmienna **kruch$** w której przechowywana jest litera odpowiadająca ostatniemu ruchowi Barnaby. Dopiszmy linię powodującą wyskoczenie z pętli w przypadku swierdzenia niezgodności zmiennej ruch$ i kruch$.

```
525   IF ruch$<>kruch$ THEN GOTO 7000
```

W linii 7000 piszemy na przykład taki komunikat:

```
7000 CLS
7010 PRINT "Pomyliles sie!"
```

Przy okazji możemy dopisać podprogram wyboru, czy gramy dalej.

```
8000 PRINT"Grasz dalej? (t/n)"
8010 INPUT odp$
8020 IF odp$="t" THEN RUN
8030 IF odp$="n" THEN END
8040 GOTO 8000
```

Właściwie gra jest już gotowa. Warto jednak wprowadzić do niej jeszcze jedno utrudnienie — upływający czas. Umieścimy więc w naszym programie jeszcze jedną zmienną — punkty, której wartość (na początku 10000) będzie się zmniejszała w miarę upływu czasu.

```
15 LET punkty=10000
515   LET punkty=punkty-1
```

W każdej prawdziwej grze komputerowej możemy na początku wybrać stopień trudności. My też damy grającemu taką możliwość. W naszym przypadku będzie to liczba ruchów do powtórzenia, oznaczona zmienną **stopień**. Wpiszemy więc następujące linie (zmieniając tym samym "starą" linię 40).

```
40 PRINT "Wybierz stopien trudnosci (1-5)."
50 INPUT stopien
60 IF stopien >5 OR stopien<1 THEN GOTO 40
70 LET stopien=2*(stopien+5)
80 CLS
```

Na koniec uzupełnimy nasz program o „ściągawkę", jakie znaczenie mają poszczególne klawisze i dopiszemy komunikat o liczbie zdobytych punktów na zakończenie gry

```
100 PRINT:PRINT
105 PRINT "                    Q"
110 PRINT
115 PRINT "                   A S"
120 PRINT
125 PRINT "                   Z X"
900 CLS
910 PRINT"Zdobyles ";punkty;" punktow."
920 GOTO 8000
```

Cóż to jednak za taniec, gdy nie słychać muzyki? Ale to już zadanie dla Was. Na każdym mikrokomputerze będzie go trzeba rozwiązać trochę inaczej. Jestem pewien, że świetnie sobie poradzicie.

*Romek*

```
10 CLS
14 REM ****** dane poczatkowe ******
15 LET punkty=10000
20 LET ruch$=""
39 REM ***** stopien trudnosci *****
40 PRINT "Wybierz stopien trudnosci (1-5)."
50 INPUT stopien
60 IF stopien >5 OR stopien<1 THEN GOTO 40
70 LET stopien=2*(stopien+5)
80 CLS
99 REM ******* opis klawiszy *******
100 PRINT:PRINT
105 PRINT "                    Q"
110 PRINT
115 PRINT "                   A S"
120 PRINT
125 PRINT "                   Z X"
149 REM ***** pozycja startowa *****
150 LOCATE 1,1
160 GOSUB 1000
170 PRINT:PRINT
180 GOSUB 1000
199 REM ******* petla glowna *******
200 FOR k=1 TO stopien
210   LOCATE 1,1
219 REM ****** ruch komputera ******
220 LET los=INT(RND(1)*5)+1
230   RESTORE
240    FOR i=1 TO los
250      READ kruch$
260    NEXT i
265   IF kruch$=ruch$ THEN GOTO 220
270   LET ruch$=kruch$
280   GOSUB 1000
499 REM ******* ruch gracza *******
500 PRINT:PRINT
510   LET ruch$=INKEY$
515   LET punkty=punkty-1
520   IF ruch$="" THEN GOTO 510
525   IF ruch$<>kruch$ THEN GOTO 7000
530   GOSUB 1000
540 NEXT k
899 REM ****** wynik koncowy ******
900 CLS
910 PRINT"Zdobyles ";punkty;" punktow."
920 GOTO 8000
999 REM ***** pozycje Kubusia *****
1000 PRINT "           %%   "
1010 PRINT "          (..)  "
1015 IF ruch$="q" THEN GOTO 2000
1020 PRINT "          (--)  "
1030 PRINT "           )(   "
1035 IF ruch$="a" THEN GOTO 3000
1036 IF ruch$="s" THEN GOTO 4000
1040 PRINT "          /XX\  "
1045 IF ruch$="z" THEN GOTO 5000
1046 IF ruch$="x" THEN GOTO 6000
1050 PRINT "          IXXI  "
1060 PRINT "          IXXI  "
1070 PRINT "           II   "
1080 PRINT "           II   "
1090 PRINT "          <II>  "
1100 RETURN
2000 PRINT "          (\/)  "
2010 GOTO 1030
3000 PRINT "         ==XX\  "
3010 PRINT "           XXI  "
3020 PRINT "           XXI  "
3030 GOTO 1070
4000 PRINT "           /XX== "
4010 PRINT "           IXX   "
4020 PRINT "           IXX   "
4030 GOTO 1070
5000 PRINT "        A IXXI   "
5010 PRINT "        ---XXI   "
5020 PRINT "            I    "
5030 PRINT "            I    "
5040 PRINT "            I>   "
5050 GOTO 1100
6000 PRINT "          IXXI A "
6010 PRINT "          IXX--- "
6020 PRINT "           I     "
6030 PRINT "           I     "
6040 PRINT "          <I     "
6050 GOTO 1100
6999 REM * komunikat o przegranej *
7000 CLS
7010 PRINT "Pomyliles sie!"
7999 REM ****** grasz dalej? ******
8000 PRINT"Grasz dalej? (t/n)"
8010 INPUT odp$
8020 IF odp$="t" THEN RUN
8030 IF odp$="n" THEN END
8040 GOTO 8000
9000 DATA "q","a","s","z","x"
```

Modyfikacje programu:

ATARI

```
1 DIM RUCH$(5)
2 DIM KRUCH$(5)
3 DIM ODP$(5)
10 PRINT CHR$(125);
80 PRINT CHR$(125);
150 POSITION 0,0
210 POSITION 0,0
220 LET LOS=INT(RND(0)*5)+1
900 PRINT CHR$(125);
9000 DATA Q,A,S,Z,X
```

COMMODORE 64, VC 20

```
10 PRINT CHR$(147);
80 PRINT CHR$(147);
150 PRINT CHR$(19);
210 PRINT CHR$(19);
510 GET RUCH$
900 PRINT CHR$(147);
```

SPECTRUM

```
150 PRINT AT 0,0
210 PRINT AT 0,0
220 LET los=INT(RND*5)+1
```

MERITUM

```
150 PRINT @1,;
210 PRINT @1,;
220 LET los=INT(RND(0)*5)+1
510 LET ruch$=INKEY$
```

# NIE TYLKO KOMPUTERY

dokończenie ze str. 32

bliźniacze bostery przyklejone do członu drugiego stopnia. Mają po cztery metry średnicy i zaopatrzone są w najpotężniejsze obecnie na świecie silniki rakietowe o ciągu 800 ton. Napędzane są płynnym paliwem, po raz pierwszy zastosowanym w radzieckiej kosmonautyce koktajlem tlenowo-wodorowęglowym.

Otulają drugi stopień o średnicy ośmiu metrów. Jego kadłub kryje cztery dysze mniejszych silników dwustutonowych, do których potężne pompy turbinowe tłoczą mieszaninę wysokoenergetyczną wodoru i tlenu (02H2). To również swego rodzaju nowość.

Koncepcja konstrukcyjna otoczenia głównej bryły pojazdu silnikami startowymi nie jest nowa. Nawiązuje do sprawdzonej wszechstronnie formuły rakiet nośnych projektowanych w czasach Korolewa, od tych które wyniosły pierwsze Sputniki, po te, użyte do transportu statków Wostok, Woschod i Sojuz. Jednak zbieżności kończą się na układzie geometrycznym. Energia jest pojazdem przełomowym także z technologicznego punktu widzenia.

Przenoszone ładunki mocuje się po bokach drugiego stopnia w specjalnych pojemnikach (jeśli jest to fracht delikatny, na przykład anteny-baterie słoneczne) lub też bez nich (elementy potężnych stacji kosmicznych). W fachowej prasie radzieckiej pisze się, że Energia jest rakietą uniwersalną również i z tego powodu, iż przystosowana jest do katapultowania w przestrzeń pozaziemską, przyszłych radzieckich ra- kietoplanów wielokrotnego użytku. Przy czym wskazuje się, że nie należy oczekiwać startu takiego promu przed rokiem 1988.

Naturalnie wszystko co nowe, prototypowe sprawia na początku pewne Niespodzianki. i tak właśnie było z dziewiczym lotem Energii. Powiedział o tym bez osłonek jeden z dyrektorów zjednoczenia Gławkosmos Stiepan Bagadiaż w rozmowie z dziennikarzem Agencji Prasowej Nowosti-Michaiłem Czernyszowem:

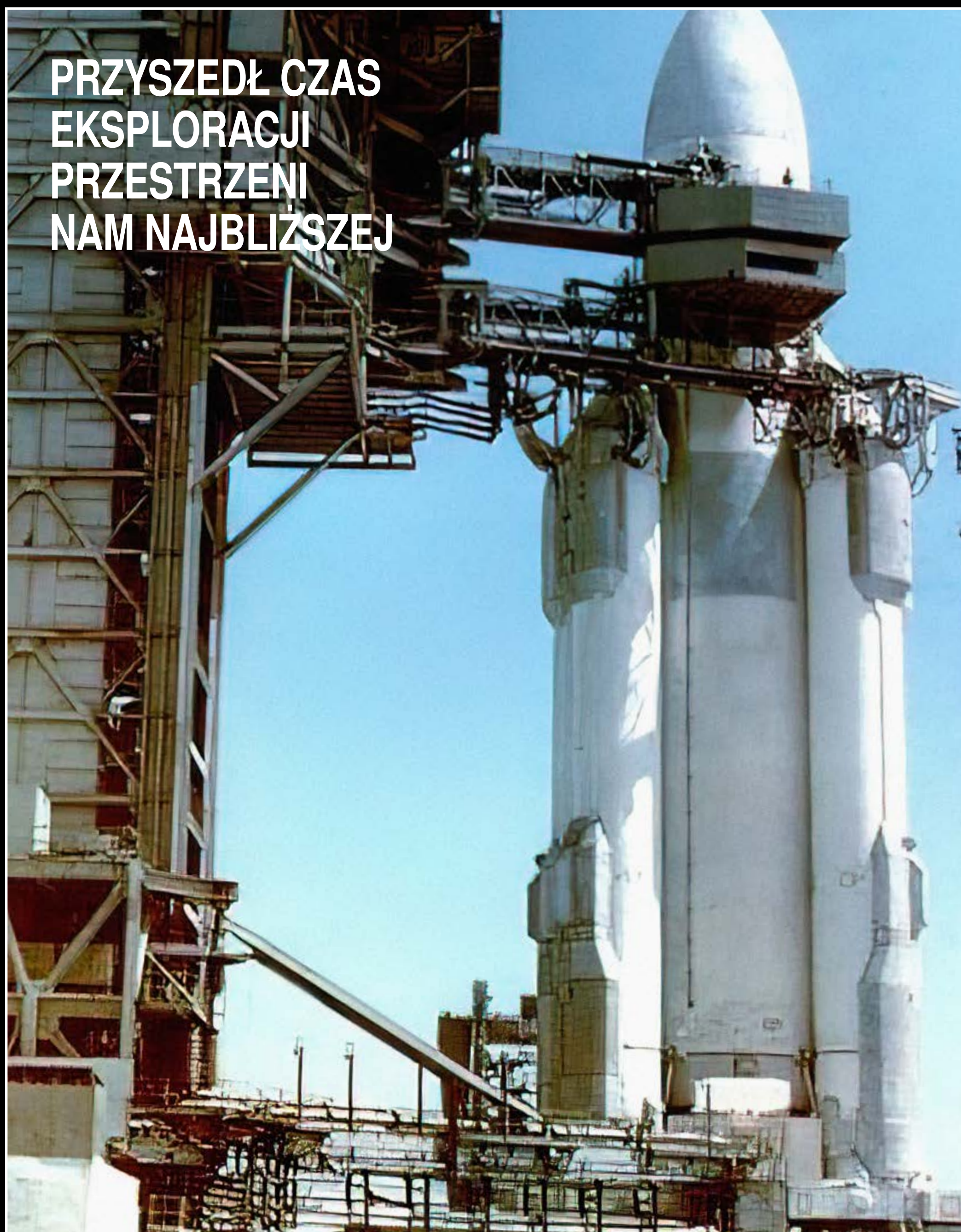
**— Pierwszy start rakiety, który miał miejsce 15 maja uważamy za udany. Oba stopnie funkcjonowały normalnie, a makieta sztucznego satelity Ziemi została wyniesiona do zaplanowanego punktu. Niestety awaria systemów pokładowych makiety stanęła na przeszkodzie wyniesieniu jej przy pomocy własnego układu silnikowego na założoną) orbitę. Na koniec makieta wylądowała w Pacyfiku. Ale ten incydent nie dotyczy samej rakiety...**

I jeszcze jeden głos, specjalisty rakietowego, uchodzącego za doskonale poinformowany, brytyjskiego tygodnika „Janes Defence Weekly", Nicholasa L. Jahnsona. Stwierdza on na wstępie, że nigdy dotąd Związek Radziecki nie odsłonił tak wysoko kurtyny przed i po starcie swego nowego statku. Tym niemniej porównuje Energię do rodziny amerykańskich rakiet nośnych Saturn V, które dość dawno przestano modernizować i wytwarzać.

Tymczasem na niedawnym kongresie kosmonautycznym, którego tematem głównym była eksploracja systemu słonecznego, dziennikarze zanotowali szokującą Zachód opinię jednego z dyrektorów programowych NASA: „Ci, którzy obawiają się transferu technologii (z Zachodu na Wschód — przyp. W.Ł.), zrobiliby lepiej obserwując jak doskonale rozwiniętymi technologiami dysponuje obecnie ZSRR. Rozumiemy, że na naszym polu może teraz powstać problem przepływu technologii do nas, a nie vice versa..."

*Wojciech Łuczak*

**15 maja bieżącego roku o 17.30 czasu środkowoeuropejskiego z jednego z radzieckich kosmodromów w Azji Centralnej wystartowała do swego dziewiczego lotu rakieta, której przyszłe regularne wyprawy na dalsze i bliższe orbity Ziemi zmienić mogą oblicze współczesnej kosmonautyki. Najnowszy superpojazd orbitalny ZSRR wzbił się w powietrze w 30 lat po wystrzeleniu w Kraju Rad pierwszego sztucznego satelity naszej planety.**



## PRZYSZEDŁ CZAS EKSPLORACJI PRZESTRZENI NAM NAJBLIŻSZEJ

Słynny Sputnik, a raczej jego prosty sygnał radiowy, doskonale słyszalny w owym czasie w każdym zakątku naszego globu, wywołał w 1957 roku oszałamiające wrażenie na całym świecie. Udane odpalenie prototypowej radzieckiej rakiety nośnej nowej generacji ochrzczonej mianem Energia nie uzyskało takiego rozgłosu. Tym niemniej w kręgach specjalistów jej próbny start miał podobny, co w przypadku Sputnika ciężar gatunkowy.

Po błyskotliwej erze spektakularnych wyczynów radzieckiej i amerykańskiej kosmonautyki przyszedł czas eksploracji przestrzeni nam najbliższej. NASA (amerykańska Agencja Aeronautyki i Przestrzeni Kosmicznej) zrezygnowała z lotów na Księżyc, ZSRR nie planuje, choć mógłby to uczynić, jak niedawno powiedział jeden z szefów radzieckich lotów załogowych, wysyłania ludzi na Marsa. Nadszedł czas przyglądania się, najlepiej ze stałych stacji, naszej starej Ziemi, która badana w ten sposób może zdradzić ludzkości jeszcze wiele swoich tajemnic. W kosmosie istnieją poza tym unikalne warunki, których stworzenie w procesach produkcji niektórych niezbędnych dzisiejszej technice materiałów, wymaga na naszej planecie skomplikowanej aparatury. A w krążącym na orbicie laboratorium idealna głęboka próżnia, czy stan nieważkością są zupełnie za darmo.

To oczywiście tylko wstęp. Marząc o dalekich wyprawach międzyplanetarnych należy najpierw skonstruować sieć baz pierwszego etapu w tak zwanym bliskim kosmosie. Na razie jedynym pojazdem, od którego można byłoby rozpocząć tworzenie kompleksów orbitalnych jest radziecka stacja Mir. Zestawianie z elementów czegoś znacznie większego wymaga wszelako sporego doświadczenia załóg, przy czym najważniejszy jest trening długotrwałego przebywania i pracy w warunkach kosmicznych, oraz odpowiedniego sprzętu transportowego zdolnego przenosić setki i tysiące ton specyficznych ładunków, a także ustawiać je w odpowiednich miejscach nad Ziemią. ZSRR od lat szkoli całe grupy kosmonautów do wielotygodniowych wypraw orbitalnych. Radzieckie doświadczenia w tej dziedzinie są niepodważalne. Teraz zaś dzięki rakiecie Energia, rozwiązano kwestię przerzutu kosmicznego frachtu.

Praktycznie od momentu katastrofy wahadłowca amerykańskiego Challenger, mówi się w USA, iż NASA popełniła kardynalny błąd skupiając się wyłącznie na promach orbitalnych. Zupełnie zrezygnowano z rozwoju jednorazowych rakiet nośnych twierdząc, że pojazdy typu space shuttle" w zupełności wystarczą. Zemściło się to na Amerykanach serią awarii pospiesznie przygotowywanych rakiet bezzato- gowych z którymi, w obliczu wstrzymania lotów wahadłowców i rosnącej liczby czekających na ekspedycję satelitów objętych różnego rodzajami zamówień, musiano się mimo wszystko przeprosić. W Stanach Zjednoczonych dopiero teraz projektuje się rakiety transportowe o podobnej co Energią klasie, również dlatego, trzeba to powiedzieć otwarcie, iż promy orbitalne nie są w stanie podołać ciężarom nałożonych na NASA zadań militarnych związanych, z programem „kosmicznej tarczy", lub jak kto woli „wojen gwiezdnych".

Do tej pory standardowymi radzieckimi kosmicznymi ciężarówkami, zdalnie sterowanymi z ośrodka kierowania lotami, były statki Progress (przeznaczone do zaopatrywania stacji kosmicznych, cumujące zwykle do jednego z sześciu węzłów modułu dokowego stacji Mir), zaś do wynoszenia i ustawiania na orbicie sztucznych satelitów Ziemi służyły i służą nadal rakiety Proton. Jak poinformowała „Prawda", a później „Izwiestia", nowa rakieta jest w stanie umieścić na orbicie geostacjonarnej obiekt o masie do 20 ton.

Już dwa lata temu, na jednym ze spotkań speców od astronautyki, radzieccy naukowcy wspominali o możliwości odpalenia pojazdu wynoszącego ładunki stutonowe. Właśnie z myślą o transporcie takiej masy zaprojektowano rakietę Energia.

Zacznijmy od tego, iż dla Energii zbudowano nową niesłychanie wytrzymałą rampę startową, na której można byłoby wznieść całe miasteczko dwunastopiętrowych domów mieszkalnych.

Monstrum ustawione pionowo na wyrzutni wyrasta w górę na 60 metrów. Pierwszy stopień (Energia jest dwustopniowa) tworzą cztery

*dokończenie na str. 31*

# ORBITALNA SUPER CIĘŻARÓWKA




www.t2e.pl

**Seria „Reduks" ma na celu przetworzenie – pełne zdigitalizowanie prasy komputerowej z początków okresu informatyzacji naszego kraju.**

W odróżnieniu od plików w formacie djvu i cbr gdzie format papierowy jest tylko zeskanowany i połączony w jeden dokument, **reduks polega na ponownym przełamaniu dokumentu, wraz z ponowną obróbką graficzną - często z ich ponowną kreacją**. Tam gdzie to możliwe zachowana jest grafika z pierwowzoru, jednak jeśli materiał jest zbyt złej jakości - zostaje on podmieniony (oczywiście jeśli jest dostępny) na równoważny. Jeżeli uda się odnaleźć błąd w zamieszczonych listingach są one poprawiane tak, aby mogły zostać uruchomione na emulatorach lub mikrokomputerach.

Rozpocząłem ten cykl od „Bajtków", jednak mam nadzieję, że uda mi się wzbogacić kolekcję zdigitalizowanych i ponownie przełamanych periodyków także o inne ważne tytuły z lat ubiegłych, o których powoli zapominamy, a bez których współczesny obraz komputeryzacji wyglądałby zgoła inaczej.

**Wydania „Reduksowe" są bezpłatne, nie roszczę sobie żadnych praw do zawartości przetwarzanych magazynów, jednak ilość czasu i pracy jaką wkładam aby mogły one trafić w Twoje ręce jest na tyle spora, że pozwolę sobie napomknąć iż możesz wesprzeć mnie przekazując dotację w dowolnej wysokości.**

Jeśli posiadasz materiały fotograficzne lub edytorskie, które są lepszej jakości niż prezentowane w numerach, daj znać.



**W ramach wolnego czasu odtwarzam stare magazyny komputerowe i prowadzę serwis emulacyjny try2emu. Jeżeli uważasz, że moja publiczna działalność możesz postawić mi kawę, a kawę lubię;)**




https://buycoffee.to/t2e

**Osoby, które wspierają serwis datkami* mają dostęp do reduksów w wysokiej rozdzielczości z usuniętymi reklamami.**

---

W zależności od wysokości datku każdy otrzymuje czasowy dostęp do reduksów w pełnej rozdzielczości na miesiąc, 3 miesiące lub 6 miesięcy.
Osoby, które aktywnie wspierają projekt mają dostęp nieograniczony.



AUTORYZOWANY SKLEP

**JEANS24h**

TUTAJ SĄ TWOJE JEANSY!

**Jeans24h.pl** - autoryzowany sklep online. Znajdziesz u nas ubrania znanych na całym świecie producentów odzieży dżinsowej – ***Lee, Wrangler, Mustang Jeans, Cross Jeans, Tom Tailor***. Kupując u nas odzież tych marek masz pewność iż pochodzą one bezpośrednio od producenta, są oryginalne i w I gatunku.

**www.jeans24h.pl**

Lee® MUSTANG® Wrangler®



CROSS JEANS®



TOM TAILOR